Tektronix®

SignalVu-PC
Vector Signal Analysis Software

Programmer Manual

www.tek.com

077-0901-02

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

Status and Events

Appendices

Glossary

Preface

The SignalVu-PC Vector Signal Analysis Software enables you to use your PC to analyze RF signal behavior with the same software used on the RSA6100B, RSA5100B, and RSA5100A Series Real-Time Analyzers. The SignalVu-PC software supports the use of the RSA6100B, RSA5100B, and RSA5100A Series commands with some limitations. For example, commands that control functions that are only available on RSA instruments.

When SignalVu-PC software is connected to an RSA306, RSA306B, RSA500A series, or RSA600A series Spectrum Analyzer, the software supports commands that control the functions of those instruments.

This manual is composed of the following sections:

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

Related Documentation

- *SignalVu-PC Vector Signal Analysis Software Help*
  The help contains detailed information about how to operate the application and is found in the SignalVu-PC application. A PDF of the Help is also available at www.tektronix.com/manuals.
- *TekVISA Programmer Manual*
  (Tektronix part number 077-0140-XX)
  This manual is available as a printable PDF file on the Tektronix Web site (www.tektronix.com/manuals). The manual describes TekVISA, the Tektronix implementation of the VISA Application Programming Interface (API). TekVISA is industry-compliant software for writing interoperable instrument drivers in a variety of Application Development Environments (ADEs).

Getting Started

Getting Started

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

- *Overview of the Manual*
  Summarizes each major section of this manual.
- *Measurement Support*
  Describes which RSA measurement displays are available with the software.
- *Setting Up Remote Communication*
  Describes how to set up TekVISA, the programming interface, and enable control over the Ethernet.

Overview of the Manual

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

Syntax and Commands

Syntax and Commands, describes the structure and content of the messages your program sends to the analyzer. The following figure shows command parts as described in the *Command Syntax* subsection.

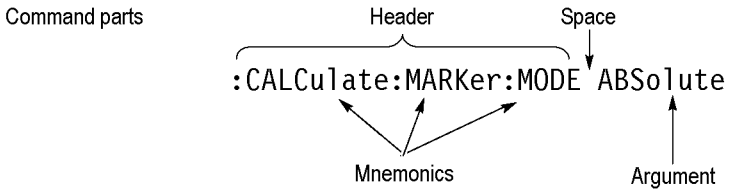


Figure 1-1: Command parts

Section 2 also describes the effect of each command and provides examples of how you might use it. The *Command Groups* subsection provides lists by functional areas. The commands are listed alphabetically in the *Command Descriptions* section.

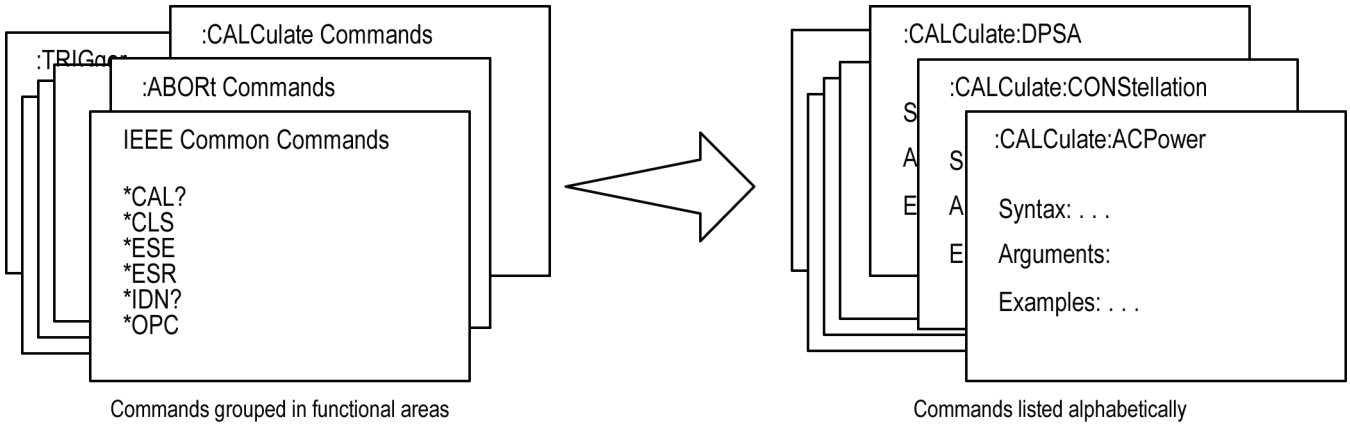


Figure 1-2: Functional groupings and an alphabetical list of commands

Status and Events

The program may request information from the instrument. The instrument provides information in the form of status and error messages. The following figure illustrates the basic operation of this system. Section 3, *Status and Events*, describes how to get status or event information from the program and details the event and error messages.

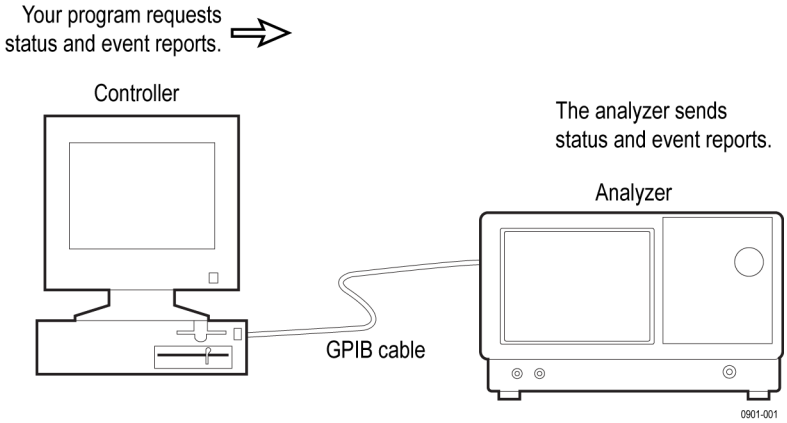


Figure 1-3: Event-driven program

Setting up remote communication

To use the SignalVu-PC programmatic interface you must install the Tektronix Virtual Instrument Software Architecture (TekVISA) on your PC. TekVISA is a Tektronix VISA solution provided for customers. It is a collection of software and documentation that allows for remote control of instruments regardless of instrument brand and physical connection. It includes a VISA driver, VISA libraries, documentation, USB Device Driver, VXI-11 server, and connection management/debug software.

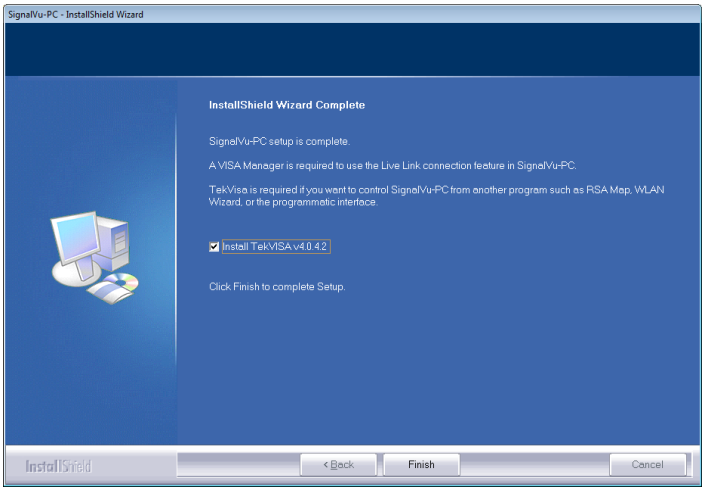
TekVISA is included as part of the SignalVu-PC application software package and you can select to automatically install it.

Setup for the programming interface

Enable the programming interface and install TekVISA. The programming interface is automatically enabled as part of the SignalVu-PC application software installation. However, in order to control SignalVu-PC using the programmatic interface, you must also install TekVISA, as follows.

The programming interface is automatically enabled as part of the SignalVu-PC application software installation. However, in order to control SignalVu-PC using the programmatic interface, you must also install TekVISA as follows.

1. Install SignalVu-PC.
2. When the installation is almost complete, the following dialog box will appear. Verify that the **Install TekVISA** box is checked.



3. Click **Finish**.

Enable control over the Ethernet

Communication with SignalVu-PC over an Ethernet LAN connection is supported with the VXI-11 protocal. The VXI-11 standard specifies an instrument protocol for TCP/IP computer networks, supporting writing and reading data to and from instruments.

The VXI-11 server is automatically installed on the target PC as part of the SignalVu-PC application software installation process. This also creates a shortcut in the Startup folder.

Right click the tray icon to set properties, check status, or start the VXI-11 or socket servers.

Measurement support

Each command works in a particular measurement view(s). Views are specified under the *Conditions* heading in the command description. The following table lists all available measurement displays supported by the SignalVu-PC software, which measurement view it is in, and if it requires an installed option.

NOTE. *If you send a command for the measurement view that is not active on screen, an execution error will occur.*

NOTE. *Some measurement displays require installed options. See the Tektronix Web site product page for a list of available options.*

Measurement support

Measurement group	Measurement
General signal viewing	Amplitude vs Time
	DPX® (Spectrum, DPXogram, and Split only)
	Frequency vs Time
	Phase vs Time
	RF I&Q vs Time
	Spectrogram
	Spectrum
	Time overview
Analog Modulation	Amplitude modulation
	Frequency modulation
	Phase modulation
Settling Time (Requires Option SV12)	Freq Settling
	Phase Settling

Measurement support (cont.)

Measurement group	Measurement
General purpose digital modulation (Requires Option SVM)	Constellation
	Demod I&Q versus Time
	EVM vs Time
	Eye Diagram
	Freq Dev vs Time
	Mag Error vs Time
	Phase Error vs Time
	Signal quality
	Symbol table
	Trellis
	User defined filters
RF	Channel power and ACPR
	Freq Settling (requires Option SVT)
	MCPR
	Occupied Bandwidth
	Phase Settling (requires Option SVT)
	Signal Strength (requires Option MAP)
	SEM
	Spurious
OFDM (Requires Option SVO)	OFDM Channel Response
	OFDM Constellation
	OFDM EVM
	OFDM Mag Error
	OFDM Phase Error
	OFDM Power
	OFDM Summary
	OFDM Symbol Table
	OFDM Flatness

Measurement support (cont.)

Measurement group	Measurement
WLAN (Requires Option SV23, SV24, or SV25)	WLAN Channel Response
	WLAN Constellation
	WLAN EVM
	WLAN Flatness
	WLAN Mag Error
	WLAN Phase Error
	WLAN Power vs Time
	WLAN Summary
	WLAN Symbol Table
	SEM
P25 Analysis (Requires Option SV26)	P25 Eye Diagram
	P25 Freq Dev vs Time
	P25 Power vs Time
	P25 Constellation
	P25 Symbol Table
	P25 Summary
	MCPR
	Time Overveiw
Bluetooth Analysis (Requires Option SV27)	BT 20dB BW
	BT CF Offset and Drift
	BT Constellation
	BT Eye Diagram
	BT Freq Dev vs Time
	BT Inband Emission
	BT Summary
	BT Symbol Table
	Time Overview
LTE Analysis (Requires Option SV28)	LTE ACLR
	LTE Channel Spectrum
	LTE Constellation
	LTE Power vs Time
	SEM
Tracking generator (Only for the RSA500A series and RSA600A series) (Requires Option 04)	Transmission Gain

Measurement support (cont.)

Measurement group	Measurement
Pulsed RF (Requires Option SVP)	Pulse statistics
	Pulse table
	Pulse trace
Audio Analysis (Requires Option SVA)	Audio Spectrum
	Audio Summary

Syntax and Commands

Command Syntax

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

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

Backus-Naur Form Definition

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

Table 2-1: BNF symbols and meanings

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

SCPI Commands and Queries

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

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

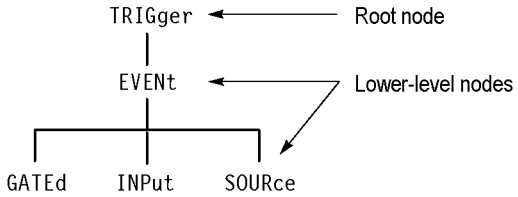


Figure 2-1: Example of SCPI subsystem hierarchy tree

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

Creating Commands

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

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

For example, TRIGgerEVENt:SOURce EXTRear is a valid SCPI command created from the hierarchy tree. (See Figure 2-1.)

Creating Queries

To create a query, start at the root node of a tree structure, move down to the end of a branch, and add a question mark. TRIGgerEVENt:SOURce? is an example of a valid SCPI query using the hierarchy tree in the figure. (See Figure 2-1.)

Query Responses

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

Table 2-2: Query response examples

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

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

Parameter Types

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

Table 2-3: Parameter types used in syntax descriptions

Parameter type	Description	Example
arbitrary block [1]	A specified length of arbitrary data	#512234xxxxx . . . where 5 indicates that the following 5 digits (12234) specify the length of the data in bytes; xxxxx ... indicates the data
boolean	Boolean numbers or values	ON or 1; OFF or 0
binary	Binary numbers	#B0110
octal	Octal numbers	#Q57, #Q3
hexadecimal [2]	Hexadecimal numbers (0-9, A, B, C, D, E, F)	#HAA, #H1
NR1 [2] numeric	Integers	0, 1, 15, -1
NR2 [2] [3] numeric	Decimal numbers	1.2, 3.141516, -6.5
NR3 [2] numeric	Floating point numbers	3.1415E-9, -16.1E5
NRf [2] numeric	Flexible decimal number that may be type NR1, NR2 or NR3	See NR1, NR2, and NR3 examples
string [4]	Alphanumeric characters (must be within quotation marks)	"Testing 1, 2, 3"

1 Defined in ANSI/IEEE 488.2 as "Definite Length Arbitrary Block Response Data."

2 An ANSI/IEEE 488.2-1992-defined parameter type.

3 Some commands and queries will accept an octal or hexadecimal value even though the parameter type is defined as NR1.

4 Defined in ANSI/IEEE 488.2 as "String Response Data."

Special Characters

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

Abbreviating Commands, Queries, and Parameters

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

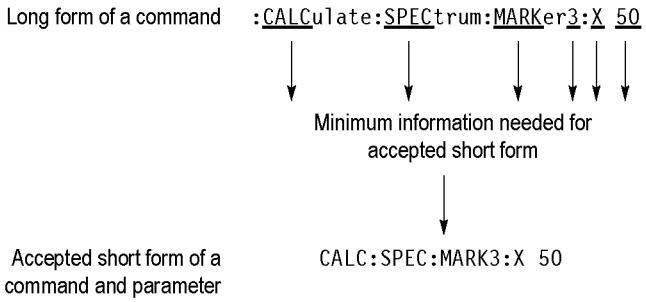


Figure 2-2: Example of abbreviating a command

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

Chaining Commands and Queries

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

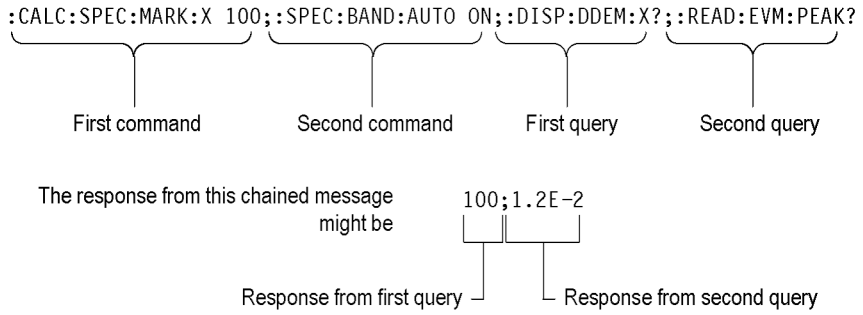


Figure 2-3: Example of chaining commands and queries

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

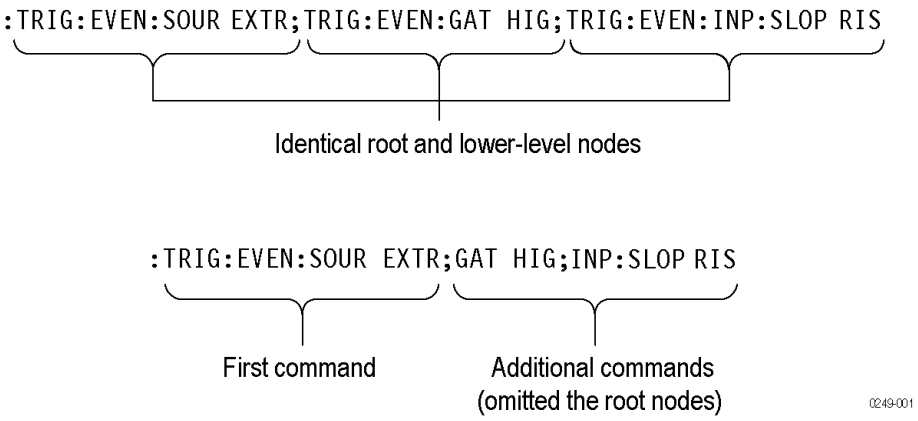


Figure 2-4: Example of omitting root and lower-level nodes in a chained message

Unit and SI Prefix

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

The following table lists the available units.

Table 2-4: Available units

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

The available SI prefixes are shown in the following table.

Table 2-5: Available SI prefixes

SI prefix	Z	A	F	P	N	U	M	K	MA[1]	G	T	PE	EX
Corresponding power	10^{-21}	10^{-18}	10^{-15}	10^{-12}	10^{-9}	10^{-6}	10^{-3}	10^{+3}	10^{+6}	10^{+9}	10^{+12}	10^{+15}	10^{+18}

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

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

15.0E6, 1.5E7Hz, 15000000, 15000000Hz, 15MHz, etc.
("15M" is not allowed.)

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

170mHz, 170mHz, 170MHz, etc.
250mv, 250mV, 250MV, etc.

General Rules

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

- You can use single (‘ ’) or double (“ ”) quotation marks for quoted strings, but you cannot use both types of quotation marks for the same string.

  correct "This string uses quotation marks correctly."

  correct ‘This string also uses quotation marks correctly.'

  incorrect "This string does not use quotation marks correctly.'

- You can use upper case, lower case, or a mixture of both cases for all commands, queries, and parameters.

  ```
  SENSE:SPECTRUM:FFT:LENGTH 1024
  ```

  is the same as

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

  and

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

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

- No embedded spaces are allowed between or within nodes.

  correct `SENSE:SPECTRUM:FFT:LENGTH 1024`

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

IEEE 488.2 Common Commands

Description

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

Command and Query Structure

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

- *ESE 16
- *CLS

The following are examples of common queries

- *ESR
- *IDN

Constructed Mnemonics

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

Table 2-6: Constructed mnemonics

Symbol	Meaning
MARKer<x>	A marker specifier where <x> = 0, 1, 2, 3, or 4. Refer to *Marker Mnemonics*
RANGe<x>	A range specifier where <x> = 1 to 20. Refer to *[SENSe]:SPURious Subgroup* for details.
SPUR<x>	A spurious specifier where <x> = 1 to the number of spurious signals. Refer to *FETCh|:READ:SPURious Subgroup* for details, respectively.
TRACe<x>	A trace specifier where <x> = 0, 1, 2, 3, 4, or 5. Refer to *TRACe Commands* for details.

Command Groups

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

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

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

For the conventions of notation in this manual, refer to *Command Syntax* and following pages.

Functional Groups

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

Table 2-7: List of command group

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

Programming Hints

Here are some basic tips for using commands:

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

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

- *Acquiring an input signal*
  Use an Initiate or Abort command to start or stop data acquisition.
  [Example] `INITiate:CONTinuous ON;INITiate:IMMediate`
  Starts data acquisition in the continuous mode.

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

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

- *Obtaining the measurement results*
  Use a Fetch or Read command to get the results.
  [Example] `FETCh:SPECtrum:TRACe1`
  Returns the spectrum trace data.

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

The following sections list the commands by group.

IEEE Common Commands

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

Table 2-8: Status and error commands

Command	Description
*CLS	Clears status.
*ESE	Sets or queries the bits in the ESER register.
*ESR?	Returns the contents of the SESR register.
*IDN?	Returns the instrument identification code.
*OPC	Synchronizes commands.
*OPT?	Returns a list of options installed in your analyzer.
*RST	Returns the instrument settings to the factory defaults.
*SRE	Sets or queries the bits in the SRER register.
*STB?	Returns the contents of the SBR using the MSS bit.
*WAI	Prevents the analyzer from executing further commands.

Abort Commands

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

Table 2-9: Abort commands

Command	Description
ABORt	Resets the trigger system and stops measurements.

Calculate Commands

Use the Calculate commands to control the markers and the search operations.

Table 2-10: Calculate commands

Command	Description
CALCulate:MARKer:ADD	Adds a marker.
CALCulate:BLUEtooth subcommand group	**Bluetooth measurements**
CALCulate:BLUEtooth:CONSte:MARKer<x>:DELTa:X[:TIME]?	Queries the delta marker time for the selected marker.
CALCulate:BLUEtooth:CONSte:MARKer<x>:FDEViation?	Queries the frequency deviation for the selected marker.
CALCulate:BLUETooth:CONSte:MARKer<x>:MAGNitude?	Queries the magnitude readout of the selected marker.
CALCulate:BLUEtooth:CONSte:MARKer<x>:MAXimum	Positions the selected marker at the symbol in the center of the analyzed time record.
CALCulate:BLUEtooth:CONSte:MARKer<x>:PEAK:LEFT	Moves the selected marker in the time domain to the next symbol number to the left, relative to the previous marker position.
CALCulate:BLUEtooth:CONSte:MARKer<x>:PEAK:RIGHt	Moves the selected marker in the time domain to the next symbol number to the right, relative to the previous marker position.
CALCulate:BLUETooth:CONSte:MARKer<x>:PHASe?	Queries the phase readout of the selected marker.
CALCulate:BLUEtooth:CONSte:MARKer<x>:SYMBol?	Queries the symbol readout value for the selected marker.
CALCulate:BLUEtooth:CONSte:MARKer<x>:Trace	Sets or queries the Trace for the selected marker.
CALCulate:BLUETooth:CONSte:MARKer<x>:VALue?	Queries the value readout of the selected marker.
CALCulate:BLUEtooth:CONSte:MARKer<x>:X	Sets or queries the time position of the selected marker.
CALCulate:BLUEtooth:EDIagram:MARKer<x>:DELTa:X[:TIME]?	Returns the delta marker time for the selected marker.
CALCulate:BLUEtooth:EDIagram:MARKer<x>:DELTa:Y?	Returns the delta marker amplitude for the selected markers.
CALCulate:BLUEtooth:EDIagram:MARKer<x>:MAXimum	Moves the selected marker to the highest peak.
CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:HIGHer	Moves the selected marker to the next peak higher in amplitude.
CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:LEFT	Moves the selected marker to the next peak to the left.
CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:LOWer	Moves the selected marker to the next peak lower in amplitude.
CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:RIGHt	Moves the selected marker to the next peak to the right.
CALCulate:BLUEtooth:EDIagram:MARKer<x>:TRACe	The command places the selected marker on the I or Q trace in the Bluetooth eye diagram display. The query returns the name of the trace where the current marker resides.
CALCulate:BLUEtooth:EDIagram:MARKer<x>:X[:TIME]	Sets or queries the horizontal position (time) of the selected marker.
CALCulate:BLUEtooth:EDIagram:MARKer<x>:Y?	Returns the vertical position of the selected marker.

Table 2-10: Calculate commands (cont.)

Command	Description
CALCulate:BLUEtooth:FDVTIME:MARKer<x>:DELTa:X[:TIME]?	Queries the delta marker time for the selected marker.
CALCulate:BLUEtooth:FDVTIME:MARKer<x>:DELTa:Y?	Returns the delta marker amplitude for the selected marker .
CALCulate:BLUEtooth:FDVTime:MARKer<x>:MAXimum	Moves the selected marker to the highest peak on the trace.
CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:HIGHer	Moves the selected marker to the next peak higher in amplitude.
CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:LEFT	Moves the selected marker to the next peak to the left on the trace .
CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:LOWer	Moves the selected marker to the next peak lower in amplitude.
CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:RIGHT	Moves the selected marker to the next peak to the right on the trace.
CALCulate:BLUEtooth:FDVTIME:MARKer<x>:X[:TIME]	Sets or queries the marker time.
CALCulate:BLUEtooth:FDVTIME:MARKer<x>:Y	Sets or queries the marker amplitude.
CALCulate:BOBW:MARKer<x>:DELTa:X?	Returns the delta marker frequency for the selected marker.
CALCulate:BOBW:MARKer<x>:DELTa:Y?	Returns the delta marker frequency for the selected marker on the Bluetooth 20 dB bandwidth measurement.
CALCulate:BOBW:MARKer<x>:MAXimum	Positions the selected marker at the maximum value of the display.
CALCulate:BOBW:MARKer<x>:PEAK:HIGHer	Moves the selected marker to the next peak higher in amplitude.
CALCulate:BOBW:MARKer<x>:PEAK:LEFT	Moves the selected marker in the time domain to the left, relative to the previous marker position on the trace.
CALCulate:BOBW:MARKer<x>:PEAK:LOWer	Moves the selected marker to the next peak lower in amplitude.
CALCulate:BOBW:MARKer<x>:PEAK:RIGHt	Moves the selected marker in the time domain to the right, relative to the previous marker position on the trace.
CALCulate:BOBW:MARKer<x>[:SET]:CENTer	Sets the center frequency to the value at the marker position.
CALCulate:BOBW:MARKer<x>:X	Sets or queries the time position of the selected marker.
CALCulate:BOBW:MARKer<x>:Y?	Returns the amplitude position of the selected marker.
CALCulate:DPX:MARKer subcommand group	**DPX power density readings**
CALCulate:DPX:MARKer<x>:POWer:DENSity?	Returns the power density of the specified marker.
CALCulate:DPX:MARKer<x>:POWer:IDensity?	Returns the integrated power density reading between the MR (reference marker) and the selected marker.
CALCulate:DPX:MARKer<x>:POWer:INTEgrated?	Returns the integrated power reading between the MR (reference marker) and the selected marker.
CALCulate:MARKer:MODE	Sets or queries the marker mode.
CALCulate:NOISe subcommand group	**Noise and Gain measurements**
CALCulate:NOISe:FIGure:MARKer<x>:DELTa:X?	Queries the frequency difference between MR (reference marker) and the selected marker.
CALCulate:NOISe:FIGure:MARKer<x>:DELTa:Y?	Queries the power difference value between MR (reference marker) and the selected marker.

Table 2-10: Calculate commands (cont.)

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

Table 2-10: Calculate commands (cont.)

Command	Description
CALCulate:NOISe:TEMPerature:MARKer<x>:TRACe	Sets or queries which trace the specified marker is attached to.
CALCulate:NOISe:TEMPerature:MARKer<x>:X?	Queries the frequency of the specified marker.
CALCulate:NOISe:TEMPerature:MARKer<x>:Y?	Queries the temperature (Kelvin) at the specified marker.
CALCulate:NOISe:YFACtor:MARKer<x>:DELTa:X?	Queries the frequency difference between MR (reference marker) and the selected marker.
CALCulate:NOISe:YFACtor:MARKer<x>:DELTa:Y?	Queries the power difference value between MR (reference marker) and the selected marker.
CALCulate:NOISe:YFACtor:MARKer<x>:MAXimum	Places the selected marker at the maximum value on the peak of the trace.
CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LEFT?	Places the selected marker at the next peak to the left on the trace.
CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LOWer?	Places the selected marker at the next lowest peak on the trace.
CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:RIGHt?	Places the specified marker on a specified trace. The query form returns the trace on which the marker resides.
CALCulate:NOISe:YFACtor:MARKer<x>:TRACe	Sets or queries which trace the specified marker is attached to.
CALCulate:NOISe:YFACtor:MARKer<x>:X?	Queries the frequency of the specified marker.
CALCulate:NOISe:YFACtor:MARKer<x>:Y?	This command queries the vertical position (dB) of the specified marker.
CALCulate:P25 subcommand group	**P25 analysis**
CALCulate:P25:CONSte:MARKer<x>:DELTa:X[:TIME]?	Returns the delta marker time for the selected marker.
CALCulate:P25:CONSte:MARKer<x>:FDEViation?	Queries the frequency deviation of the selected marker.
CALCulate:P25:CONSte:MARKer<x>:MAGNitude?	Queries the frequency deviation (for Freq Dev trace types) or magnitude (for IQ trace types) of the selected marker.
CALCulate:P25:CONSte:MARKer<x>:MAXimum	Moves the selected marker to the highest peak on the trace.
CALCulate:P25:CONSte:MARKer<x>:PEAK:LEFT	Moves the selected marker to the next peak to the left on the trace.
CALCulate:P25:CONSte:MARKer<x>:PEAK:RIGHt	Moves the selected marker to the next peak to the right on the trace.
CALCulate:P25:CONSte:MARKer<x>:PHASe?	Queries the phase of the selected marker.
CALCulate:P25:CONSte:MARKer<x>:SYMBol?	Queries the readout symbol of the selected marker.
CALCulate:P25:CONSte:MARKer<x>:Trace	Places the selected marker on the Demodulated I and Q versus Time trace. The query returns the name of the trace on which the marker resides.
CALCulate:P25:CONSte:MARKer<x>:VALue?	Queries the readout value of the specified marker.
CALCulate:P25:CONSte:MARKer<x>:X	Sets or queries the horizontal position of the selected marker.
CALCulate:P25:EDIagram:MARKer<x>:DELTa:X[:TIME]?	Returns the delta marker time for the selected marker.
CALCulate:P25:EDIagram:MARKer<x>:DELTa:Y?	Returns the delta marker amplitude for the selected marker.
CALCulate:P25:EDIagram:MARKer<x>:MAXimum	Moves the selected marker to the highest peak.
CALCulate:P25:EDIagram:MARKer<x>:PEAK:HIGHer	Moves the selected marker to the next highest peak.
CALCulate:P25:EDIagram:MARKer<x>:PEAK:LEFT	Moves the selected marker to the next peak to the left .
CALCulate:P25:EDIagram:MARKer<x>:PEAK:LOWer	Moves the selected marker to the next lowest peak.
CALCulate:P25:EDIagram:MARKer<x>:PEAK:RIGHt	Moves the selected marker to the next peak to the right .

Table 2-10: Calculate commands (cont.)

Command	Description
CALCulate:P25:EDIagram:MARKer<x>:TRACe	Sets the selected marker on the I or Q trace, or queries the trace that the specified marker is on.
CALCulate:P25:EDIagram:MARKer<x>:X	Sets or queries the horizontal position of the selected marker.
CALCulate:P25:EDIagram:MARKer<x>:Y	Sets or queries the vertical position of the selected marker.
CALCulate:P25:PVTime:MARKer<x>:DELTa:X[:TIME]?	Returns the delta marker time (seconds) for the selected marker.
CALCulate:P25:PVTime:MARKer<x>:DELTa:Y[:TIME]?	Returns the Delta marker amplitude (dB) for the selected marker.
CALCulate:P25:PVTime:MARKer<x>:Maximum	Moves the selected marker to the highest peak on the trace.
CALCulate:P25:PVTime:MARKer<x>:PEAK:HIGHer	Moves the selected marker to the next peak higher in amplitude on the trace.
CALCulate:P25:PVTime:MARKer<x>:PEAK:LEFT	Moves the selected marker to the next peak to the left on the trace.
CALCulate:P25:PVTime:MARKer<x>:PEAK:LOWer	Moves the selected marker to the next peak lower in amplitude on the trace.
CALCulate:P25:PVTime:MARKer<x>:PEAK:RIGHt	Moves the selected marker to the next peak to the right on the trace.
CALCulate:P25:PVTime:MARKer<x>:X	Sets or queries the horizontal position of the selected marker.
CALCulate:P25:PVTime:MARKer<x>:Y	Sets or queries the vertical position of the selected marker.
CALCulate: Marker subgroup	**Marker position and values**
CALCulate:SPECtrum:MARKer<x>:POWer:DENSity?	Returns the power density of the specified marker.
CALCulate:SPECtrum:MARKer<x>:POWer:IDENSity	Returns the integrated power density reading between the MR (reference marker) and the selected marker.
CALCulate:SPECtrum:MARKer<x>:POWer:INTEgrated?	Returns the integrated power reading between the MR (reference marker) and the selected marker.
CALCulate:MARKer:AOFF	Turns off all markers.
CALCulate:MARKer:DELete	Deletes the last marker added.
CALCulate:MARKer:DENSity:EXCursion	Sets or queries the minimum excursion of DPX signal density.
CALCulate:MARKer:DENSity:SMOothing	Sets or queries the number of pixels squared for smoothing the density.
CALCulate:MARKer:DENSity:THReshold	Sets or queries the threshold of DPX signal density to detect peaks.
CALCulate:MARKer:DRAG:SEARch:STATe	Enables or disables peak searching when dragging a marker in a plot.
CALCulate:MARKer:MODE	Sets or queries the marker mode.
CALCulate:MARKer:PEAK:EXCursion	Sets or queries the minimum excursion level.
CALCulate:MARKer:PEAK:THReshold	Sets or queries the threshold level to detect peaks.
CALCulate:SEARch subgroup	**Calculate search commands**
CALCulate:SEARch:LIMit:FAIL?	Queries whether the waveform cuts across the limit or not.
CALCulate:SEARch:LIMit:MATCh:BEEP[:STATe]	Sets or queries whether to beep when a match occurs.
CALCulate:SEARch:LIMit:MATCh:SACQuire[:STATe]	Sets or queries whether to stop acquiring data on match.
CALCulate:SEARch:LIMit:MATCh:SDATa[:STATe]	Sets or queries whether to save the acquisition data automatically.
CALCulate:SEARch:LIMit:MATCh:SPICture[:STATe]	Sets or queries whether to save the whole screen automatically.
CALCulate:SEARch:LIMit:MATCh:STRace[:STATe]	Sets or queries whether to save the spectrum trace automatically.
CALCulate:SEARch:LIMit:OPERation	Sets or queries the limit operation in the search function.

Table 2-10: Calculate commands (cont.)

Command	Description
CALCulate:SEARch:LIMit:OPERation:FEED	Sets or queries the data flow to be fed in the search operation.
CALCulate:SEARch:LIMit:OPERation:MASK:LOAD	Loads the limit mask from a specified file for the search operation.
CALCulate:SEARch:LIMit:OPERation:MASK:STORE	Stores the limit mask to a specified file for the search operation.
CALCulate:SEARch:LIMit:OPERation:SLIMit	Sets or queries the limit value in the search operation.
CALCulate:SEARch:LIMit:REPort:DATA?	Returns the frequency range(s) that satisfy the search condition.
CALCulate:SEARch:LIMit:REPort:POINts?	Returns the number of range(s) that satisfy the search condition.
CALCulate:SEARch:LIMit:STATe	Sets or queries whether to enable or disable the search function.
CALCulate:ACPower subgroup	**Channel power and ACPR measurement**
CALCulate:ACPower:MARKer<x>:DELTa:X?	Returns the delta marker frequency for the selected marker.
CALCulate:ACPower:MARKer<x>:DELTa:Y?	Returns the delta marker amplitude for the selected marker.
CALCulate:ACPower:MARKer<x>:MAXimum	Moves the marker to the highest peak on the trace.
CALCulate:ACPower:MARKer<x>:PEAK:LEFT	Moves the marker to the next peak to the left on the trace.
CALCulate:ACPower:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right on the trace.
CALCulate:ACPower:MARKer<x>:X	Sets or queries the horizontal position of the marker.
CALCulate:ACPower:MARKer<x>:Y?	Queries the marker amplitude of the selected marker.
CALCulate:{AM|FM|PM} subgroup (Option 21 only)	**AM/FM/PM measurement**
CALCulate:{AM|FM|PM}:MARKer<x>:DELTa:X?	Returns the delta marker time for the selected marker.
CALCulate:{AM|FM|PM}:MARKer<x>:DELTa:Y?	Returns the delta marker amplitude for the selected marker.
CALCulate:{AM|FM|PM}:MARKer<x>:MAXimum	Moves the marker to the highest peak on the trace.
CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:HIGHer	Moves the marker to the next peak higher in amplitude.
CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:LEFT	Moves the marker to the next peak to the left on the trace.
CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:LOWer	Moves the marker to the next peak lower in amplitude.
CALCulate:{AM|FM|PM}:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right on the trace.
CALCulate:{AM|FM|PM}:MARKer<x>:X	Sets or queries the horizontal position (time) of the marker.
CALCulate:{AM|FM|PM}:MARKer<x>:Y?	Queries the marker amplitude of the selected marker.
CALCulate:AUDio subgroup	**Audio measurements**
CALCulate:AUDio:HARMonic:HNUMber	Sets or queries the number of harmonics.
CALCulate:AUDio:HARMonic:NHNumber	Sets or queries the number of non-harmonics.
CALCulate:AUDio:NHARmonic:EXCursion	Sets or queries the non-harmonic excursion value.
CALCulate:AUDio:NHARmonic:IGNore	Sets or queries the ignore region of the non-harmonics.
CALCulate:AUDio:NHARmonic:THReshold	Sets or queries the non-harmonic threshold value.
CALCulate:AUDio:SPECtrum:MARKer<x>:DELTa:X?	Returns the delta marker frequency for the selected marker on the spectrum trace.
CALCulate:AUDio:SPECtrum:MARKer<x>:DELTa:Y?	Returns the delta marker amplitude for the selected marker on the spectrum trace.
CALCulate:AUDio:SPECtrum:MARKer<x>:MAXimum	Moves the selected marker to the highest peak on the spectrum trace.

Table 2-10: Calculate commands (cont.)

Command	Description
CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:HIGHer	Moves the selected marker to the next peak higher in amplitude on the spectrum trace.
CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:LEFT	Moves the selected marker to the next peak to the left on the spectrum trace.
CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:LOWer	Moves the selected marker to the next peak lower in amplitude on the spectrum trace.
CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:RIGHt	Moves the selected marker to the next peak to the right on the spectrum trace.
CALCulate:AUDio:SPECtrum:MARKer<x>:TRACe	Sets or queries the trace on which the specified marker is placed in the spectrum measurement.
CALCulate:AUDio:SPECtrum:MARKer<x>:X	Sets or queries the horizontal position of the selected marker on the spectrum trace.
CALCulate:AUDio:SPECtrum:MARKer<x>:Y?	Queries the marker amplitude of the selected marker on the spectrum trace.
CALCulate:AVTime subgroup	**Frequency versus Time measurement**
CALCulate:AVTime:MARKer<x>:DELTa:X?	Returns the delta marker time for the selected marker.
CALCulate:AVTime:MARKer<x>:DELTa:Y?	Returns the delta marker amplitude for the selected marker.
CALCulate:AVTime:MARKer<x>:MAXimum	Moves the marker to the highest peak on the trace.
CALCulate:AVTime:MARKer<x>:PEAK:HIGHer	Moves the marker to the next peak higher in amplitude.
CALCulate:AVTime:MARKer<x>:PEAK:LEFT	Moves the marker to the next peak to the left on the trace.
CALCulate:AVTime:MARKer<x>:PEAK:LOWer	Moves the marker to the next peak lower in amplitude.
CALCulate:AVTime:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right on the trace.
CALCulate:AVTime:MARKer<x>:TRACe	Sets or queries the trace on which the marker is placed.
CALCulate:AVTime:MARKer<x>:X	Sets or queries the horizontal position of the marker.
CALCulate:AVTime:MARKer<x>:Y?	Queries the marker amplitude of the selected marker.
CALCulate:CONSte subgroup (Option 21 only)	**Constellation measurement**
CALCulate:CONSte:MARKer<x>:DELTa:X[:TIME]?	Returns the delta marker time for the selected marker.
CALCulate:CONSte:MARKer<x>:FDEViation?	Queries the frequency deviation of the selected marker.
CALCulate:CONSte:MARKer<x>:MAGNitude?	Queries the magnitude readout of the marker.
CALCulate:CONSte:MARKer<x>:MAXimum	Positions the marker at the symbol in the center of the time record.
CALCulate:CONSte:MARKer<x>:PEAK:LEFT	Moves the marker in the time domain to the next lower symbol number.
CALCulate:CONSte:MARKer<x>:PEAK:RIGHt	Moves the marker in the time domain to the next higher symbol number.
CALCulate:CONSte:MARKer<x>:PHASe?	Queries the phase readout of the marker.
CALCulate:CONSte:MARKer<x>:SYMBol?	Queries the symbol readout of the marker.
CALCulate:CONSte:MARKer<x>:VALue?	Queries the value readout of the marker.
CALCulate:CONSte:MARKer<x>:X	Sets or queries the time position of the marker on the trace.
CALCulate:DIQVtime subgroup (Option 21 only)	**Demodulated I&Q versus Time measurement**
CALCulate:DIQVtime:MARKer<x>:DELTa:X[:TIME]?	Returns the delta marker time for the selected marker.

Table 2-10: Calculate commands (cont.)

Command	Description
CALCulate:DIQVtime:MARKer<x>:DELTa:Y?	Returns the delta marker amplitude for the selected marker.
CALCulate:DIQVtime:MARKer<x>:MAXimum	Moves the marker to the highest peak on the trace.
CALCulate:DIQVtime:MARKer<x>:PEAK:HIGHer	Moves the marker to the next peak higher in amplitude.
CALCulate:DIQVtime:MARKer<x>:PEAK:LEFT	Moves the marker to the next peak to the left on the trace.
CALCulate:DIQVtime:MARKer<x>:PEAK:LOWer	Moves the marker to the next peak lower in amplitude.
CALCulate:DIQVtime:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right on the trace.
CALCulate:DIQVtime:MARKer<x>:TRACe	Places the selected marker on the Demodulated I&Q versus Time trace.
CALCulate:DIQVtime:MARKer<x>:X[:TIME]	Sets or queries the horizontal position (time) of the marker.
CALCulate:DIQVtime:MARKer<x>:Y?	Queries the marker amplitude of the selected marker.
CALCulate:DPX subgroup	**DPX spectrum measurement**
CALCulate:DPX:MARKer<x>:DELTa:Y?	Returns the delta marker amplitude for the selected marker.
CALCulate:DPX:MARKer<x>:MAXimum	Moves the marker to the highest peak on the trace.
CALCulate:DPX:MARKer<x>:PEAK:HIGHer	Moves the marker to the next peak higher in amplitude.
CALCulate:DPX:MARKer<x>:PEAK:LEFT	Moves the marker to the next peak to the left on the trace.
CALCulate:DPX:MARKer<x>:PEAK:LOWer	Moves the marker to the next peak lower in amplitude.
CALCulate:DPX:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right on the trace.
CALCulate:DPX:MARKer<x>[:SET]:CENTer	Sets the center frequency to the marker frequency.
CALCulate:DPX:MARKer<x>:TRACe	Sets or queries the trace to attach the marker to.
CALCulate:DPX:MARKer<x>:X:AMPLitude	Sets or queries the amplitude position of the marker.
CALCulate:DPX:MARKer<x>:X[:FREQuency]	Sets or queries the frequency position of the marker.
CALCulate:DPX:MARKer<x>:X:PHASe	Sets or queries the phase position of the selected marker.
CALCulate:DPX:MARKer<x>:X:TIME	Sets or queries the time position of the selected marker.
CALCulate:DPX:MARKer<x>:Y?	Queries the marker amplitude of the selected marker.
CALCulate:EDIagram subgroup (Option 21 only)	**Eye diagram**
CALCulate:EDIagram:MARKer<x>:DELTa:X[:TIME]?	Returns the delta marker time for the selected marker.
CALCulate:EDIagram:MARKer<x>:DELTa:Y?	Returns the delta marker amplitude for the selected marker.
CALCulate:EDIagram:MARKer<x>:MAXimum	Moves the marker to the highest peak on the trace.
CALCulate:EDIagram:MARKer<x>:PEAK:HIGHer	Moves the marker to the next peak higher in amplitude.
CALCulate:EDIagram:MARKer<x>:PEAK:LEFT	Moves the marker to the next peak to the left on the trace.
CALCulate:EDIagram:MARKer<x>:PEAK:LOWer	Moves the marker to the next peak lower in amplitude.
CALCulate:EDIagram:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right on the trace.
CALCulate:EDIagram:MARKer<x>:TRACe	Places a marker on the I or Q trace in the eye diagram display.
CALCulate:EDIagram:MARKer<x>:X[:TIME]	Sets or queries the horizontal position (time) of the marker.
CALCulate:EDIagram:MARKer<x>:Y?	Queries the marker amplitude of the selected marker.
CALCulate:EVM subgroup (Option 21 only)	**EVM versus Time measurement**
CALCulate:EVM:MARKer<x>:DELTa:X?	Returns the delta marker time for the selected marker.
CALCulate:EVM:MARKer<x>:DELTa:Y?	Returns the delta marker amplitude for the selected marker.

Table 2-10: Calculate commands (cont.)

Command	Description
CALCulate:EVM:MARKer<x>:MAXimum	Moves the marker to the highest peak on the trace.
CALCulate:EVM:MARKer<x>:PEAK:HIGHer	Moves the marker to the next peak higher in amplitude.
CALCulate:EVM:MARKer<x>:PEAK:LEFT	Moves the marker to the next peak to the left on the trace.
CALCulate:EVM:MARKer<x>:PEAK:LOWer	Moves the marker to the next peak lower in amplitude.
CALCulate:EVM:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right on the trace.
CALCulate:EVM:MARKer<x>:X	Sets or queries the horizontal position of the marker.
CALCulate:EVM:MARKer<x>:Y?	Queries the marker amplitude of the selected marker.
CALCulate:FDVTime subgroup (Option 21 only)	**Frequency deviation versus Time measurement**
CALCulate:FDVTime:MARKer<x>:DELTa:X[:TIME]?	Returns the delta marker time for the selected marker.
CALCulate:FDVTime:MARKer<x>:DELTa:Y?	Returns the delta marker frequency for the selected marker.
CALCulate:FDVTime:MARKer<x>:MAXimum	Moves the marker to the highest peak on the trace.
CALCulate:FDVTime:MARKer<x>:PEAK:HIGHer	Moves the marker to the next peak higher in amplitude.
CALCulate:FDVTime:MARKer<x>:PEAK:LEFT	Moves the marker to the next peak to the left on the trace.
CALCulate:FDVTime:MARKer<x>:PEAK:LOWer	Moves the marker to the next peak lower in amplitude.
CALCulate:FDVTime:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right on the trace.
CALCulate:FDVTime:MARKer<x>:X[:TIME]	Sets or queries the horizontal position (time) of the marker.
CALCulate:FDVTime:MARKer<x>:Y?	Queries the marker amplitude of the selected marker.
CALCulate:{FSETtling|PSETtling} subgroup	**Frequency and Phase Settling measurements**
CALCulate:{FSETtling|PSETtling}:MARKer<x>:DELTa:X?	Returns the delta marker time for the selected marker in the Frequency and Phase Settling displays.
CALCulate:{FSETtling|PSETtling}:MARKer<x>:DELTa:Y?	Returns the delta marker frequency for the selected marker in the Frequency and Phase Settling displays.
CALCulate:{FSETtling|PSETtling}:MARKer<x>:MAXimum	Moves the selected marker to the highest peak on the trace in the Frequency or Phase Settling display.
CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:HIGHer	Moves the selected marker to the next peak higher in amplitude on the Frequency or Phase Settling trace.
CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:LEFT	Moves the selected marker to the next peak to the left on the trace in the Frequency or Phase Settling display.
CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:LOWer	Moves the selected marker to the next peak lower in amplitude on the Frequency or Phase Settling trace.
CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:RIGHt	Moves the selected marker to the next peak to the right on the trace in the Frequency or Phase Settling display.
CALCulate:{FSETtling|PSETtling}:MARKer<x>:TRACe	Sets or queries the trace on which the specified marker is placed in the spectrum measurement
CALCulate:{FSETtling|PSETtling}:MARKer<x>:X	Sets or queries the horizontal position of the selected marker in the Frequency and Phase Settling display.
CALCulate:{FSETtling|PSETtling}:MARKer<x>:Y?	Queries the marker amplitude of the selected marker in the Frequency and Phase Settling display.

Table 2-10: Calculate commands (cont.)

Command	Description
CALCulate:FVTime subgroup	**Frequency versus Time measurement**
CALCulate:FVTime:MARKer<x>:DELTa:X?	Returns the delta marker time for the selected marker.
CALCulate:FVTime:MARKer<x>:DELTa:Y?	Returns the delta marker frequency for the selected marker.
CALCulate:FVTime:MARKer<x>:MAXimum	Moves the marker to the highest peak on the trace.
CALCulate:FVTime:MARKer<x>:PEAK:HIGHer	Moves the marker to the next peak higher in amplitude.
CALCulate:FVTime:MARKer<x>:PEAK:LEFT	Moves the marker to the next peak to the left on the trace.
CALCulate:FVTime:MARKer<x>:PEAK:LOWer	Moves the marker to the next peak lower in amplitude.
CALCulate:FVTime:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right on the trace.
CALCulate:FVTime:MARKer<x>:X	Sets or queries the horizontal position of the marker.
CALCulate:FVTime:MARKer<x>:Y?	Queries the marker amplitude of the selected marker.
CALCulate:IQVTime subgroup	**RF I&Q versus Time measurement**
CALCulate:IQVTime:MARKer<x>:DELTa:X?	Returns the delta marker time for the selected marker.
CALCulate:IQVTime:MARKer<x>:DELTa:Y?	Returns the delta marker amplitude for the selected marker.
CALCulate:IQVTime:MARKer<x>:MAXimum	Moves the marker to the highest peak on the trace.
CALCulate:IQVTime:MARKer<x>:PEAK:HIGHer	Moves the marker to the next peak higher in amplitude.
CALCulate:IQVTime:MARKer<x>:PEAK:LEFT	Moves the marker to the next peak to the left on the trace.
CALCulate:IQVTime:MARKer<x>:PEAK:LOWer	Moves the marker to the next peak lower in amplitude.
CALCulate:IQVTime:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right on the trace.
CALCulate:IQVTime:MARKer<x>:TRACe	Sets or queries the trace (I or Q) to place the marker.
CALCulate:IQVTime:MARKer<x>:X	Sets or queries the horizontal position of the marker.
CALCulate:IQVTime:MARKer<x>:Y?	Queries the marker amplitude of the selected marker.
CALCulate:MCPower subgroup	**MCPR measurement**
CALCulate:MCPower:MARKer<x>:DELTa:X?	Returns the delta marker frequency for the selected marker.
CALCulate:MCPower:MARKer<x>:DELTa:Y?	Returns the delta marker amplitude for the selected marker.
CALCulate:MCPower:MARKer<x>:MAXimum	Moves the marker to the highest peak on the trace.
CALCulate:MCPower:MARKer<x>:PEAK:LEFT	Moves the marker to the next peak to the left on the trace.
CALCulate:MCPower:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right on the trace.
CALCulate:MCPower:MARKer<x>:X	Sets or queries the horizontal position of the marker.
CALCulate:MCPower:MARKer<x>:Y?	Queries the marker amplitude of the selected marker.
CALCulate:MERRor subgroup (Option 21 only)	**Magnitude error versus Time measurement**
CALCulate:MERRor:MARKer<x>:DELTa:X?	Returns the delta marker time for the selected marker.
CALCulate:MERRor:MARKer<x>:DELTa:Y?	Returns the delta marker amplitude for the selected marker.
CALCulate:MERRor:MARKer<x>:MAXimum	Moves the marker to the highest peak on the trace.
CALCulate:MERRor:MARKer<x>:PEAK:HIGHer	Moves the marker to the next peak higher in amplitude.
CALCulate:MERRor:MARKer<x>:PEAK:LEFT	Moves the marker to the next peak to the left on the trace.
CALCulate:MERRor:MARKer<x>:PEAK:LOWer	Moves the marker to the next peak lower in amplitude.
CALCulate:MERRor:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right on the trace.

Table 2-10: Calculate commands (cont.)

Command	Description
CALCulate:MERRor:MARKer<x>:X	Sets or queries the horizontal position of the marker.
CALCulate:MERRor:MARKer<x>:Y?	Queries the marker amplitude of the selected marker.
CALCulate:OBWidth subgroup	**Occupied Bandwidth measurement**
CALCulate:OBWidth:MARKer<x>:DELTa:X?	Returns the delta marker frequency for the selected marker.
CALCulate:OBWidth:MARKer<x>:DELTa:Y?	Returns the delta marker amplitude for the selected marker.
CALCulate:OBWidth:MARKer<x>:MAXimum	Moves the marker to the highest peak on the trace.
CALCulate:OBWidth:MARKer<x>:PEAK:HIGHer	Moves the marker to the next peak higher in amplitude.
CALCulate:OBWidth:MARKer<x>:PEAK:LEFT	Moves the marker to the next peak to the left on the trace.
CALCulate:OBWidth:MARKer<x>:PEAK:LOWer	Moves the marker to the next peak lower in amplitude.
CALCulate:OBWidth:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right on the trace.
CALCulate:OBWidth:MARKer<x>[:SET]:CENTer	Sets the center frequency to the value at the marker position.
CALCulate:OBWidth:MARKer<x>:X	Sets or queries the horizontal position of the marker.
CALCulate:OBWidth:MARKer<x>:Y?	Queries the marker amplitude of the selected marker.
CALCulate:OFDM subgroup	**Orthogonal frequency division multiplexing measurement**
CALCulate:OFDM:CONSte:MARKer<x>:FREQuency	Sets or queries the frequency position of the selected marker.
CALCulate:OFDM:CONSte:MARKer<x>:MAGNitude?	Queries the marker magnitude of the selected marker.
CALCulate:OFDM:CONSte:MARKer<x>:PHASe?	Queries the phase readout of the selected marker.
CALCulate:OFDM:CONSte:MARKer<x>:TIME	Sets or queries the X position of the selected marker.
CALCulate:OFDM:CONSte:MARKer<x>:TYPE?	Queries the data type of the selected marker.
CALCulate:OFDM:CONSte:MARKer<x>:VALue?	Queries the value of the specified marker.
CALCulate:OFDM:CRESponse:MARKer<x>:DELTa:X?	Returns the delta marker frequency for the selected marker.
CALCulate:OFDM:CRESponse:MARKer<x>:DELTa:Y?	Returns the delta marker frequency for the selected marker.
CALCulate:OFDM:CRESponse:MARKer<x>:MAXimum	Moves the selected marker to the highest peak.
CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:HIGHer	Moves the selected marker to the next peak higher in amplitude.
CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:LEFT	Moves the selected marker to the next peak to the left.
CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:LOWer	Moves the selected marker to the next peak lower in amplitude.
CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:RIGHt	Moves the selected marker to the next peak to the right.
CALCulate:OFDM:CRESponse:MARKer<x>:X	Sets or queries the horizontal position of the selected marker.
CALCulate:OFDM:CRESponse:MARKer<x>:Y?	Returns the amplitude of the selected marker.
CALCulate:OFDM:FLATness:MARKer<x>:MAXimum	Positions the specified marker at the maximum point on the trace.
CALCulate:OFDM:FLATness:MARKer<x>:PEAK:HIGHer	Moves the specified marker to the next peak higher in amplitude on the trace.
CALCulate:OFDM:FLATness:MARKer<x>:PEAK:LEFT	Moves the specified marker to the next peak to the left on the trace.
CALCulate:OFDM:FLATness:MARKer<x>:PEAK:LOWer	Moves the marker to the next peak lower in amplitude on the trace.
CALCulate:OFDM:FLATness:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right on the trace.

Table 2-10: Calculate commands (cont.)

Command	Description
CALCulate:OFDM:FLATness:MARKer<x>:X	Sets or queries the horizontal value at the selected marker position on the points trace.
CALCulate:OFDM:FLATness:MARKer<x>:Y?	Returns the value of the amplitude (vertical position) at the selected marker position.
CALCulate:OFDM:TABLe:MARKer<x>:FREQuency	Sets or queries the frequency position of the selected marker.
CALCulate:OFDM:TABLe:MARKer<x>:TIMe	Sets or queries the marker time.
CALCulate:OFDM:TABLe:MARKer<x>:VALue?	Queries the value readout of the selected marker.
CALCulate:PERRor subgroup (Option 21 only)	**Phase error versus Time measurement**
CALCulate:PERRor:MARKer<x>:DELTa:X?	Returns the delta marker time for the selected marker.
CALCulate:PERRor:MARKer<x>:DELTa:Y?	Returns the delta marker phase for the selected marker.
CALCulate:PERRor:MARKer<x>:MAXimum	Moves the marker to the highest peak on the trace.
CALCulate:PERRor:MARKer<x>:PEAK:HIGHer	Moves the marker to the next peak higher in amplitude.
CALCulate:PERRor:MARKer<x>:PEAK:LEFT	Moves the marker to the next peak to the left on the trace.
CALCulate:PERRor:MARKer<x>:PEAK:LOWer	Moves the marker to the next peak lower in amplitude.
CALCulate:PERRor:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right on the trace.
CALCulate:PERRor:MARKer<x>:X	Sets or queries the horizontal position of the marker.
CALCulate:PERRor:MARKer<x>:Y?	Queries the marker amplitude of the selected marker.
CALCulate:PHVTime subgroup	**Phase versus Time measurement**
CALCulate:PHVTime:MARKer<x>:DELTa:X?	Returns the delta marker time for the selected marker.
CALCulate:PHVTime:MARKer<x>:DELTa:Y?	Returns the delta marker phase for the selected marker.
CALCulate:PHVTime:MARKer<x>:MAXimum	Moves the marker to the highest peak on the trace.
CALCulate:PHVTime:MARKer<x>:PEAK:HIGHer	Moves the marker to the next peak higher in amplitude.
CALCulate:PHVTime:MARKer<x>:PEAK:LEFT	Moves the marker to the next peak to the left on the trace.
CALCulate:PHVTime:MARKer<x>:PEAK:LOWer	Moves the marker to the next peak lower in amplitude.
CALCulate:PHVTime:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right on the trace.
CALCulate:PHVTime:MARKer<x>:X	Sets or queries the horizontal position of the marker.
CALCulate:PHVTime:MARKer<x>:Y?	Queries the marker amplitude of the selected marker.
CALCulate:PULSe subgroup	**Pulsed RF measurements**
CALCulate:PULSe:STATistics:FFT:INDicator:X	Sets or queries the frequency in Hz for the X cursor.
CALCulate:PULSe:STATistics:FFT:INDicator:Y?	Returns the frequency level in dB for the Y cursor.
CALCulate:PULSe:STATistics:HISTogram:INDicator:X	Sets or queries the histogram indicator X value.
CALCulate:PULSe:STATistics:HISTogram:INDicator:Y?	Queries the histogram indicator Yvalue.
CALCulate:PULSe:STATistics:MARKer<x>:DELTa:X?	Returns the delta marker frequency for the specified marker.
CALCulate:PULSe:STATistics:MARKer<x>:DELTa:Y?	Returns the delta marker amplitude for the specified marker.
CALCulate:PULSe:STATistics:MARKer<x>:MAXimum	Moves the marker to the highest peak on the trace.
CALCulate:PULSe:STATistics:MARKer<x>:PEAK:HIGHer	Moves the marker to the next peak higher in amplitude.
CALCulate:PULSe:STATistics:MARKer<x>:PEAK:LEFT	Moves the marker to the next peak to the left on the statistics trace.

Table 2-10: Calculate commands (cont.)

Command	Description
CALCulate:PULSe:STATistics:MARKer<x>:PEAK:LOWer	Moves the marker to the next peak lower in amplitude.
CALCulate:PULSe:STATistics:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right on the statistics trace.
CALCulate:PULSe:STATistics:MARKer<x>:X	Sets or queries the horizontal position of the marker.
CALCulate:PULSe:STATistics:MARKer<x>:Y?	Queries the marker amplitude of the selected marker.
CALCulate:PULSe:TRACe:MARKer<x>:DELTa:X?	Returns the delta marker time for the selected marker.
CALCulate:PULSe:TRACe:MARKer<x>:DELTa:Y?	Returns the delta marker amplitude for the selected marker.
CALCulate:PULSe:TRACe:MARKer<x>:MAXimum	Moves the marker to the highest peak on the pulse trace.
CALCulate:PULSe:TRACe:MARKer<x>:PEAK:HIGHer	Moves the marker to the next peak higher in amplitude.
CALCulate:PULSe:TRACe:MARKer<x>:PEAK:LEFT	Moves the marker to the next peak to the left on the pulse trace.
CALCulate:PULSe:TRACe:MARKer<x>:PEAK:LOWer	Moves the marker to the next peak lower in amplitude.
CALCulate:PULSe:TRACe:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right on the pulse trace.
CALCulate:PULSe:TRACe:MARKer<x>:X	Sets or queries the horizontal position of the marker.
CALCulate:PULSe:TRACe:MARKer<x>:Y?	Queries the marker amplitude of the selected marker.
CALCulate:SEARch subgroup	**Spectrogram measurement**
CALCulate:SEARch:LIMit:FAIL?	Queries whether the waveform cuts across the limit in the search operation.
CALCulate:SEARch:LIMit:MATCh:BEEP[:STATe]	Sets or queries whether or not to emit a beep on match during run or replay in the search operation.
CALCulate:SEARch:LIMit:MATCh:SACQuire[:STATe]	Sets or queries whether or not to stop acquiring data on match during run or replay in the search operation.
CALCulate:SEARch:LIMit:MATCh:SDATa[:STATe]	Sets or queries whether or not to save automatically (AutoSave) acquisition data on match during run in the search operation.
CALCulate:SEARch:LIMit:MATCh:SPICture[:STATe]	Sets or queries whether or not to save automatically (AutoSave) the whole screen on match during run in the search operation.
CALCulate:SEARch:LIMit:MATCh:STRace[:STATe]	Sets or queries whether or not to save automatically (AutoSave) the spectrum trace on match during run in the search operation.
CALCulate:SEARch:LIMit:OPERation	Sets or queries the search condition parameter.
CALCulate:SEARch:LIMit:OPERation:FEED	Sets or queries the data flow to be fed in the search operation.
CALCulate:SEARch:LIMit:OPERation:MASK:LOAD	Loads the specified limit mask file for the search operation.
CALCulate:SEARch:LIMit:OPERation:MASK:STORE	Saves the limit mask to a specified file.
CALCulate:SEARch:LIMit:OPERation:SLIMit	Sets or queries the limit value (in dB) in the search operation.
CALCulate:SEARch:LIMit:REPort:DATA?	Queries the frequency range(s) that satisfy the search condition.
CALCulate:SEARch:LIMit:REPort:POINts?	Queries the number of frequency range(s) that satisfy the search condition.
CALCulate:SEARch:LIMit:STATe	Sets or queries the search function (enabled or disabled).
CALCulate:SEM subgroup	**Spectral Emissions Mask measurement**
CALCulate:SEM:MARKer<x>:DELTa:X?	Returns the delta marker frequency for the selected marker.
CALCulate:SEM:MARKer<x>:DELTa:Y?	Returns the delta marker amplitude for the selected marker.

Table 2-10: Calculate commands (cont.)

Command	Description
CALCulate:SEM:MARKer<x>:MAXimum	Moves the selected marker to the highest peak.
CALCulate:SEM:MARKer<x>:PEAK:HIGHer	Moves the selected marker to the next peak higher in amplitude.
CALCulate:SEM:MARKer<x>:PEAK:LEFT	Moves the selected marker to the next peak to the left.
CALCulate:SEM:MARKer<x>:PEAK:LOWer	Moves the selected marker to the next peak lower in amplitude.
CALCulate:SEM:MARKer<x>:PEAK:RIGHt	Moves the selected marker to the next peak to the right.
CALCulate:SEM:MARKer<x>[:SET]:CENTer	Moves the specified marker to the center frequency.
CALCulate:SEM:MARKer<x>:X	Sets or queries the horizontal position of the selected marker.
CALCulate:SEM:MARKer<x>:Y?	Queries the marker amplitude of the selected marker.
CALCulate:SGRam subgroup	**Spectrogram measurement**
CALCulate:SGRam:MARKer<x>:DELTa:X:FREQuency?	Returns the delta marker frequency for the selected marker.
CALCulate:SGRam:MARKer<x>:DELTa:X[:TIME]?	Returns the delta marker time for the selected marker.
CALCulate:SGRam:MARKer<x>:DELTa:Y?	Returns the delta marker amplitude for the selected marker.
CALCulate:SGRam:MARKer<x>:MAXimum	Moves the marker to the highest peak on the trace.
CALCulate:SGRam:MARKer<x>:PEAK:HIGHer	Moves the marker to the next peak higher in amplitude.
CALCulate:SGRam:MARKer<x>:PEAK:LEFT	Moves the marker to the next peak to the left on the trace.
CALCulate:SGRam:MARKer<x>:PEAK:LOWer	Moves the marker to the next peak lower in amplitude.
CALCulate:SGRam:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right on the trace.
CALCulate:SGRam:MARKer<x>[:SET]:CENTer	Sets the center frequency to the marker frequency.
CALCulate:SGRam:MARKer<x>:X:FREQuency	Sets or queries the marker frequency.
CALCulate:SGRam:MARKer<x>:X[:TIME]	Sets or queries the marker time.
CALCulate:SGRam:MARKer<x>:Y?	Queries the marker amplitude of the selected marker.
CALCulate:SPECtrum subgroup	**Spectrum measurement**
CALCulate:SPECtrum:MARKer<x>:DELTa:X?	Returns the delta marker frequency for the selected marker.
CALCulate:SPECtrum:MARKer<x>:DELTa:Y?	Returns the delta marker amplitude for the selected marker.
CALCulate:SPECtrum:MARKer<x>:MAXimum	Moves the marker to the highest peak on the trace.
CALCulate:SPECtrum:MARKer<x>:PEAK:HIGHer	Moves the marker to the next peak higher in amplitude.
CALCulate:SPECtrum:MARKer<x>:PEAK:LEFT	Moves the marker to the next peak to the left on the trace.
CALCulate:SPECtrum:MARKer<x>:PEAK:LOWer	Moves the marker to the next peak lower in amplitude.
CALCulate:SPECtrum:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right on the trace.
CALCulate:SPECtrum:MARKer<x>[:SET]:CENTer	Sets the center frequency to the marker frequency.
CALCulate:SPECtrum:MARKer<x>:TRACe	Sets or queries the trace on which the marker is placed.
CALCulate:SPECtrum:MARKer<x>:X	Sets or queries the horizontal position of the marker.
CALCulate:SPECtrum:MARKer<x>:Y?	Queries the marker amplitude of the selected marker.
CALCulate:SPURious subgroup	**Spurious measurement**
CALCulate:SPURious:MARKer<x>:DELTa:X?	Returns the delta marker frequency for the selected marker.
CALCulate:SPURious:MARKer<x>:DELTa:Y?	Returns the delta marker amplitude for the selected marker.
CALCulate:SPURious:MARKer<x>:MAXimum	Moves the marker to the highest peak on the trace.

Table 2-10: Calculate commands (cont.)

Command	Description
CALCulate:SPURious:MARKer<x>:PEAK:HIGHer	Moves the marker to the next peak higher in amplitude.
CALCulate:SPURious:MARKer<x>:PEAK:LEFT	Moves the marker to the next peak to the left on the trace.
CALCulate:SPURious:MARKer<x>:PEAK:LOWer	Moves the marker to the next peak lower in amplitude.
CALCulate:SPURious:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right on the trace.
CALCulate:SPURious:MARKer<x>[:SET]:CENTer	Sets the center frequency to the marker frequency.
CALCulate:SPURious:MARKer<x>:X	Sets or queries the horizontal position of the marker.
CALCulate:SPURious:MARKer<x>:Y?	Queries the marker amplitude of the selected marker.
CALCulate:TDIagram subgroup (Option 21 only)	**Trellis diagram**
CALCulate:TDIagram:MARKer<x>:DELTa:X[:TIME]?	Returns the delta marker time for the selected marker.
CALCulate:TDIagram:MARKer<x>:DELTa:Y?	Returns the delta marker amplitude for the selected marker.
CALCulate:TDIagram:MARKer<x>:MAXimum	Moves the marker to the highest peak on the trace.
CALCulate:TDIagram:MARKer<x>:PEAK:HIGHer	Moves the marker to the next peak higher in amplitude.
CALCulate:TDIagram:MARKer<x>:PEAK:LEFT	Moves the marker to the next peak to the left on the trace.
CALCulate:TDIagram:MARKer<x>:PEAK:LOWer	Moves the marker to the next peak lower in amplitude.
CALCulate:TDIagram:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right on the trace.
CALCulate:TDIagram:MARKer<x>:X[:TIME]	Sets or queries the horizontal position (time) of the marker.
CALCulate:TDIagram:MARKer<x>:Y?	Queries the marker amplitude of the selected marker.
CALCulate:TOVerview subgroup	**Time overview**
CALCulate:TOVerview:MARKer<x>:DELTa:X?	Returns the delta marker time for the selected marker.
CALCulate:TOVerview:MARKer<x>:DELTa:Y?	Returns the delta marker amplitude for the selected marker.
CALCulate:TOVerview:MARKer<x>:MAXimum	Moves the marker to the highest peak on the trace.
CALCulate:TOVerview:MARKer<x>:PEAK:HIGHer	Moves the marker to the next peak higher in amplitude.
CALCulate:TOVerview:MARKer<x>:PEAK:LEFT	Moves the marker to the next peak to the left on the trace.
CALCulate:TOVerview:MARKer<x>:PEAK:LOWer	Moves the marker to the next peak lower in amplitude.
CALCulate:TOVerview:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right on the trace.
CALCulate:TOVerview:MARKer<x>:X	Sets or queries the horizontal position of the marker.
CALCulate:TOVerview:MARKer<x>:Y?	Queries the marker amplitude of the selected marker.
CALCulate:WLAN subgroup	**Wireless LAN measurement**
CALCulate:WLAN:CONSte:MARKer<x>:FREQuency	Sets or queries the frequency value of the marker.
CALCulate:WLAN:CONSte:MARKer<x>:MAGNitude?	Returns the magnitude value at the indicated marker position.
CALCulate:WLAN:CONSte:MARKer<x>:PHASe?	Returns the phase value at the indicated marker position.
CALCulate:WLAN:CONSte:MARKer<x>:TIME	Sets the time offset value at the selected marker position on the points trace.
CALCulate:WLAN:CONSte:MARKer<x>:TYPE?	Returns the data type of the subcarrier for data at the indicated marker position.
CALCulate:WLAN:CONSte:MARKer<x>:VALue?	Queries the value readout at the selected marker position on the points trace.

Table 2-10: Calculate commands (cont.)

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

Table 2-10: Calculate commands (cont.)

Command	Description
CALCulate:WLAN:MERRor:MARKer<x>:DELTa:SUBCarrier?	Queries the delta Subcarrier or frequency value at the selected marker position.
CALCulate:WLAN:MERRor:MARKer<x>:DELTa:SYMBol?	Queries the delta time value in seconds or symbols at the selected marker position.
CALCulate:WLAN:MERRor:MARKer<x>:DELTa:Y?	Queries the vertical delta value for the points trace at the selected marker position.
CALCulate:WLAN:MERRor:MARKer<x>:FREQuency	Sets or queries the frequency value of the marker.
CALCulate:WLAN:MERRor:MARKer<x>:TIME	Sets the time offset value at the selected marker position.
CALCulate:WLAN:MERRor:MARKer<x>:VALue?	Queries the value readout at the selected marker position.
CALCulate:WLAN:PERRor:MARKer<x>:AVERage:SUBCarrier?	Queries the Phase Error value at the selected marker position on the average trace.
CALCulate:WLAN:PERRor:MARKer<x>:AVERage:SYMBol?	Queries the Phase Error value at the selected marker position on the average trace.
CALCulate:WLAN:PERRor:MARKer<x>:DELTa:SUBCarrier?	Queries the delta Subcarrier or frequency value at the selected marker position.
CALCulate:WLAN:PERRor:MARKer<x>:DELTa:SYMBol?	Queries the delta time value in seconds or symbols at the selected marker position.
CALCulate:WLAN:PERRor:MARKer<x>:DELTa:Y?	Queries the vertical delta value for the points trace at the selected marker position.
CALCulate:WLAN:PERRor:MARKer<x>:FREQuency	Sets or queries the frequency value at the selected marker position.
CALCulate:WLAN:PERRor:MARKer<x>:TIME	Sets the time value (offset) at the selected marker position.
CALCulate:WLAN:PERRor:MARKer<x>:VALue?	Queries the Phase Error value at the selected marker position.
CALCulate:WLAN:PVTime:MARKer<x>:DELTa:X?	Returns the delta time value at the selected marker position.
CALCulate:WLAN:PVTime:MARKer<x>:DELTa:Y?	Queries the vertical delta value for the points trace at the selected marker position.
CALCulate:WLAN:PVTime:MARKer<x>:MAXimum	Positions the specified marker at the maximum point.
CALCulate:WLAN:PVTime:MARKer<x>:PEAK:HIGHer	Moves the specified marker to the next peak higher in amplitude.
CALCulate:WLAN:PVTime:MARKer<x>:PEAK:LEFT	Moves the specified marker to the next peak on the left.
CALCulate:WLAN:PVTime:MARKer<x>:PEAK:LOWer	Moves the marker to the next peak lower in amplitude.
CALCulate:WLAN:PVTime:MARKer<x>:PEAK:RIGHt	Moves the marker to the next peak to the right.
CALCulate:WLAN:PVTime:MARKer<x>:X	Sets or queries the value of the horizontal position for the selected marker
CALCulate:WLAN:PVTime:MARKer<x>:Y?	Returns the amplitude at the selected marker position.
CALCulate:WLAN:TABLe:MARKer<x>:FREQuency	Assigns or queries the frequency value of the marker.
CALCulate:WLAN:TABLe:MARKer<x>:TIME	Sets or queries the vertical value at the selected marker position.
CALCulate:WLAN:TABLe:MARKer<x>:VALue?	Queries the decoded symbol value at the selected marker position.
CALCulate:LTE subgroup	
CALCulate:LTE:ACLR:MARKer:<x>PEAK:HIGHer	Moves the specified marker to the next peak higher in amplitude on the trace.

Table 2-10: Calculate commands (cont.)

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

Marker Mnemonics

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

Table 2-11: Marker mnemonics

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

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

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

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

Calibration Commands

Use the CALibration commands to control the external correction.

Not all of these commands apply to SignalVu-PC and connected instruments. See the individual command descriptions for details.

Table 2-12: Calibration commands

Command	Description
CALibration:ABORt	Aborts any actions related to the alignments in progress.
CALibration:TXGain:ABORt	
CALibration:TXGain[:ALL]	
CALibration:TXGain:FINish?	
CALibration:TXGain:STATe?	
CALibration:AUTO	Sets or queries whether or not to run alignments automatically.
CALibration:CORRection:EXTernal:EDIT<x>:LABel	Sets or queries the name of the external loss table.
CALibration:CORRection:EXTernal:EDIT<x>:NEW	Creates a new external loss table.
CALibration:CORRection:EXTernal:EDIT<x>:STATe	Sets or queries whether to enable or disable the external loss table.
CALibration:CORRection:EXTernal:GAIN[:MAGNitude]	Sets or queries the external gain value.
CALibration:CORRection:EXTernal:GAIN:STATe	Sets or queries whether to enable or disable the external gain value.
CALibration:CORRection:EXTernal:PROBe:CONNect?	Queries whether the external probe is connected to the analyzer or not.
CALibration:CORRection:EXTernal:PROBe[:MAGNitude]?	Queries the external probe attenuation.
CALibration:CORRection:EXTernal:PROBe:STATe	Determines whether to correct data for the external probe attenuation.
CALibration:CORRection:EXTernal:TYPE	Selects data type to use when applying the external loss table.

Display Commands

Use the DISPlay commands to control the display of measurement waveforms and results on the screen.

Table 2-13: Display commands

Command	Description
DISPlay Noise subgroup	**Noise Figure and Gain measurements**
DISPlay:WINDow:ACTive:MEASurement?	Queries the active measurement views.
DISPlay Bluetooth subgroup	**Bluetooth measurements**
DISPlay:BIBEmissions:MARKer<x>:SHOW:STATe	Sets or queries to show or hide the readout for the selected marker.
DISPlay:BIBEmissions:PLEVel:SHOW:STATe	Sets or queries to show or hide the power levels
DISPlay:BIBEmissions:RESet:SCALe	Resets the horizontal and vertical scale to the default values.
DISPlay:BIBEmissions:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries the graticule hidden or showing state.
DISPlay:BIBEmissions:X[:SCALe]	Sets or queries the horizontal range of the graph.
DISPlay:BIBEmissions:X[:SCALe]:AUTO	Rescales the horizontal axis automatically to fit the waveform to the screen in the Bluetooth InBand Emission view.
DISPlay:BIBEmissions:X[:SCALe]:OFFSet	Sets or queries the center frequency.
DISPlay:BIBEmissions:Y[:SCALe]	Sets or queries the vertical range of the graph.
DISPlay:BIBEmissions:Y[:SCALe]:AUTO	Rescales the vertical axis automatically to fit the waveform to the screen in the Bluetooth InBand Emission view.
DISPlay:BIBEmissions:Y[:SCALe]:OFFSet	Sets or queries the vertical offset.
DISPlay:BLUEtooth:CONSte:TRACe:GRATicule:GRID:STATe	Sets or queries the display state of graticule grid.
DISPlay:BLUEtooth:CONSte:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries whether to show or hide the graticule grid on the screen in the display.
DISPlay:BLUEtooth:EDIagram:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries graticule grid hidden or showing state in the Bluetooth eye diagram display.
DISPlay:BLUEtooth:EDIagram:Y[:SCALe]	Sets or queries the vertical range.
DISPlay:BLUEtooth:EDIagram:Y[:SCALe]:AUTO	Sets the vertical scale automatically to fit the waveform in the Bluetooth eye diagram.
DISPlay:BLUEtooth:EDIagram:Y[:SCALe]:OFFSet	Sets or queries the vertical offset (center point of the vertical axis).
DISPlay:BLUEtooth:FDVTime:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries graticule grid hidden or showing state in the Freq Dev vs Time display.
DISPlay:BLUEtooth:FDVTime:Y[:SCALe]:AUTO	Rescales the vertical scale automatically to fit the waveform to the screen.
DISPlay:BLUEtooth:FDVTime:Y[:SCALe]:OFFSet	Sets or queries the vertical offset (center point of the vertical axis).
DISPlay:BLUEtooth:MEASview:DELete	Deletes the specified Bluetooth analysis view.
DISPlay:BLUEtooth:MEASview:NEW	Displays a new measurement view for Bluetooth measurements.
DISPlay:BLUEtooth:MEASview:SELect	Selects a measurement view on the screen in the Bluetooth measurements. The query command returns the currently selected view.

Table 2-13: Display commands (cont.)

Command	Description
DISPlay:BOBW:SELected:BANDwidth	Sets or queries the bandwidth to measure in the Bluetooth 20 dB BW view.
DISPlay:DPX:WINDow:TRACe:GRATicule:GRID:STATe	Determines whether to show or hide the graticule grid on the DPX Spectrum display..
DISPlay:NOISe:FIGure[:SCALe]:AUTO	Automatically scales the graph.
DISPlay:NOISe:FIGure:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries the graticule state to on (showing) or off (hidden).
DISPlay:NOISe:FIGure:WINDow:TRACe:LEGend:STATe	Sets or queries the trace legend state to on (showing) or off (hidden).
DISPlay:NOISe:FIGure:WINDow:TRACe:MEASpoints:STATe	Sets or queries the measurement points state to on (showing) or off (hidden).
DISPlay:NOISe:FIGure:X:AUTO	Automatically sets the horizontal scale of the graph.
DISPlay:NOISe:FIGure:X:OFFSet	Sets or queries the center frequency (offset).
DISPlay:NOISe:FIGure:X[:SCALe]	Sets or queries the horizontal scale of the display.
DISPlay:NOISe:FIGure:Y:AUTO	Automatically sets the vertical scale of the graph.
DISPlay:NOISe:FIGure:Y:AUTO:STATe	Sets the state of the automatic vertical scale and position of the graph to on or off. The query form of the command returns the current state.
DISPlay:NOISe:FIGure:Y:PDIVision	Sets or queries the distance between graticule lines on the display.
DISPlay:NOISe:FIGure:Y:POSition	Sets or queries the vertical position of the display.
DISPlay:NOISe:FIGure:Y:SCALe	Sets or queries the vertical scale of the display.
DISPlay:NOISe:FIGure:Y:SCALe:LINear:STATe	Sets or queries whether the linear units are shown (on) or not shown (off) on the display.
DISPlay:NOISe:GAIN[:SCALe]:AUTO	Automatically scales the graph.
DISPlay:NOISe:GAIN:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries the graticule state to on (showing) or off (hidden).
DISPlay:NOISe:GAIN:WINDow:TRACe:LEGend:STATe	Sets or queries the trace legend state to on (showing) or off (hidden).
DISPlay:NOISe:GAIN:WINDow:TRACe:MEASPoints:STATe	Sets or queries the measurement points state to on (showing) or off (hidden).
DISPlay:NOISe:GAIN:X:AUTO	Automatically sets the horizontal scale of the display.
DISPlay:NOISe:GAIN:X:OFFSet	Sets or queries the center frequency (offset).
DISPlay:NOISe:GAIN:X:SCALe	Sets or queries the horizontal scale of the display.
DISPlay:NOISe:GAIN:Y:AUTO	Automatically sets the vertical scale of the graph.
DISPlay:NOISe:GAIN:Y:AUTO:STATe	Sets or queries the state of the automatic vertical scale and position of the graph to on or off.
DISPlay:NOISe:GAIN:Y:PDIVision	Sets or queries the distance between graticule lines on the display.
DISPlay:NOISe:GAIN:Y:POSition	Sets or queries the vertical position of the display.
DISPlay:NOISe:GAIN:Y:SCALe	Sets or queries the vertical scale of the display.
DISPlay:NOISe:MEASview:DELete	Deletes the selected Noise and Gain measurement view.
DISPlay:NOISe:MEASview:NEW	Displays a new Noise and Gain measurement view.

Table 2-13: Display commands (cont.)

Command	Description
DISPlay:NOISe:MEASview:SELect	Selects a noise measurement view. The query command returns the currently selected view.
DISPlay:NOISe:POWer:LINear:STATe	Sets or queries whether the linear units are shown (on) or not shown (off) on the Noise Table display.
DISPlay:NOISe:TEMPerature[:SCALe]:AUTO	Automatically scales the graph.
DISPlay:NOISe:TEMPerature:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries the graticule state to on (showing) or off (hidden).
DISPlay:NOISe:TEMPerature:WINDow:TRACe:LEGend:STATe	Sets or queries the trace legend state to on (showing) or off (hidden).
DISPlay:NOISe:TEMPerature:WINDow:TRACe:MEASPoints:STATe	Sets or queries the measurement points state to on (showing) or off (hidden).
DISPlay:NOISe:TEMPerature:X:AUTO	Automatically sets the horizontal scale of the graph.
DISPlay:NOISe:TEMPerature:X:OFFSet	Sets or queries the center frequency (offset) value.
DISPlay:NOISe:TEMPerature:X[:SCALe]	Sets or queries the horizontal scale of the graph.
DISPlay:NOISe:TEMPerature:X:STARt	Sets or queries the Start Frequency value.
DISPlay:NOISe:TEMPerature:X:STOP	Sets or queries the Stop Frequency value.
DISPlay:NOISe:TEMPerature:Y:AUTO	Automatically sets the vertical scale of the graph.
DISPlay:NOISe:TEMPerature:Y:AUTO:STATe	Sets or queries the state of the automatic vertical scale and position of the graph to on or off.
DISPlay:NOISe:TEMPerature:Y:PDIVision	Sets or queries the distance between graticule lines on the display.
DISPlay:NOISe:TEMPerature:Y:POSition	Sets or queries the vertical position of the graph.
DISPlay:NOISe:TEMPerature:Y:SCALe	Sets or queries the vertical scale of the graph.
DISPlay:NOISe:UNCERTainty:RESult:GAIN?	Queries the gain computed uncertainty value.
DISPlay:NOISe:UNCERTainty:RESult:NFIGure?	Queries the noise figure computed uncertainty value.
DISPlay:NOISe:YFACtor[:SCALe]:AUTO	Automatically scales the Y Factor graph.
DISPlay:NOISe:YFACtor:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries the graticule state to on (showing) or off (hidden).
DISPlay:NOISe:YFACtor:WINDow:TRACe:LEGend:STATe	Sets or queries the trace legend state to on (showing) or off (hidden).
DISPlay:NOISe:YFACtor:WINDow:TRACe:MEASPoints:STATe	Sets or queries the measurement points state to on (showing) or off (hidden).
DISPlay:NOISe:YFACtor:X:AUTO	Automatically sets the horizontal scale.
DISPlay:NOISe:YFACtor:X:OFFSet	Sets or queries the center frequency (offset).
DISPlay:NOISe:YFACtor:X[:SCALe]	Sets or queries the horizontal scale.
DISPlay:NOISe:YFACtor:X:STARt	Sets or queries the Start Frequency value.
DISPlay:NOISe:YFACtor:X:STOP	Sets or queries the Stop Frequency value.
DISPlay:NOISe:YFACtor:Y:AUTO	Automatically sets the vertical scale of the graph.
DISPlay:NOISe:YFACtor:Y:AUTO:STATe	Sets or queries the automatic vertical scale and position of the graph setting to on or off.

Table 2-13: Display commands (cont.)

Command	Description
DISPlay:NOISe:YFACtor:Y:PDIVision	Sets or queries the distance between graticule lines on the display in dB.
DISPlay:NOISe:YFACtor:Y:POSition	Sets or queries the vertical position.
DISPlay:NOISe:YFACtor:Y:SCALe	Sets or queries the vertical scale.
DISPlay:P25:CONSte:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries to show or hide the graticule grid on the screen.
DISPlay:P25:EDIagram:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries to show or hide the graticule grid on the screen.
DISPlay:P25:EDIagram:Y[:SCALe]	Sets or queries the vertical scale.
DISPlay:P25:EDIagram:Y[:SCALe]:AUTO	Sets the vertical scale automatically to fit the waveform to the screen.
DISPlay:P25:EDIagram:Y[:SCALe]:OFFSet	Sets or queries the vertical offset (center point of the vertical axis).
DISPlay:P25:MEASview:DELete	Deletes the specified P25 measurement display.
DISPlay:P25:MEASview:NEW	Displays a new P25 measurement view.
DISPlay:P25:MEASview:SELect	Selects a P25 measurement display. The query form returns the currently selected display.
DISPlay:P25:PVTime:BURSt:X[:SCALe]	Sets or queries the value of the scale (width) value, in seconds, when using horizontal Full Burst view.
DISPlay:P25:PVTime:BURSt:X[:SCALe]:AUTO	Automatically sets the starting time (position) and scale (width) values for the best display in the horizontal Full Burst view.
DISPlay:P25:PVTime:BURSt:X[:SCALe]:OFFSet	Sets or queries the starting time (position) value, in seconds, when using the horizontal Full Burst view.
DISPlay:P25:PVTime:BURSt:X[:SCALe]:RESet	Resets the starting time (position) and scale (width) values for the horizontal Full Burst view.
DISPlay:P25:PVTime:FALL:X[:SCALe]	Sets or queries the scale (width) value, in seconds, when using the horizontal Falling Edge view.
DISPlay:P25:PVTime:FALL:X[:SCALe]:AUTO	Automatically sets the starting time (position) and scale (width) values for the best display in the horizontal Falling Edge view.
DISPlay:P25:PVTime:FALL:X[:SCALe]:OFFSet	Sets or queries the value for the starting time (offset) of the graph, in seconds, when using the horizontal Falling Edge view.
DISPlay:P25:PVTime:FALL:X[:SCALe]:RESet	Resets the starting time (position) and scale (width) values for the horizontal Falling Edge view.
DISPlay:P25:PVTime:MARKer:SHOW:STATe	Shows or hides the marker readout.
DISPlay:P25:PVTime:RISE:X[:SCALe]	Sets or queries the scale (width) value, in seconds, for the P25 Power vs. Time display when using the horizontal Rising Edge view.
DISPlay:P25:PVTime:RISE:X[:SCALe]:AUTO	Sets the scale (width) value, in seconds, to automatic.
DISPlay:P25:PVTime:RISE:X[:SCALe]:OFFSet	Sets or queries the starting time value (offset), in seconds, for the P25 Power vs. Time display when using the horizontal Rising Edge view.
DISPlay:P25:PVTime:RISE:X[:SCALe]:RESet	Resets the starting time (position) and scale (width) values for the horizontal Rising Edge view.
DISPlay:P25:PVTime:WINDow:SELect:PLOT	Sets or queries which view to use: Full Burst, Rising Edge, or Falling Edge.

Table 2-13: Display commands (cont.)

Command	Description
DISPlay:P25:PVTime:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries the graticule state to on (showing) or off (hidden).
DISPlay:P25:PVTime:Y[:SCALe]	Sets or queries the vertical scale value.
DISPlay:P25:PVTime:Y[:SCALe]:AUTO	Automatically selects the vertical scale and position values.
DISPlay:P25:PVTime:Y[:SCALe]:OFFSet	Sets or queries the value of the vertical offset (top edge of the vertical axis).
DISPlay:P25:PVTime:Y[:SCALe]:PDIVision	Sets or queries the value of the vertical scale for the P25 Power vs. Time display.
DISPlay:P25:RADix	Set or queries the symbol radix for the P25 Symbol Table display.
DISPlay:TOVerview:WINDow:NAVigator:STATe	Sets or queries whether the navigator view of the Time Overview display is on or off.
DISPlay:TOVerview:WINDow:TIME:MODE	Sets or queries the type of time analysis to be performed.
DISPlay:TOVerview:WINDow:TRACe:LEGend:STATe	Sets or queries the trace legend state to on (showing) or off (hidden).
DISPlay basic command subgroup	**General window control**
DISPlay:WINDow:COLor:SCHeme	Sets or queries the color scheme for traces and background.
DISPlay:WINDow:OPTimized:MEASurement?	Queries the measurement views that are optimized.
DISPlay:ACPower subgroup	**Channel power and ACPR measurement**
DISPlay:ACPower:MARKer:SHOW:STATe	Determines whether to show the readout for the selected marker.
DISPlay:ACPower:PLEVel:SHOW:STATe	Determines whether to show the power levels.
DISPlay:ACPower:RESet:SCALe	Resets the horizontal and vertical scale to the default values.
DISPlay:ACPower:WINDow:TRACe:GRATicule:GRID:STATe	Determines whether to show the graticule grid on screen.
DISPlay:ACPower:X[:SCALe]	Sets or queries the horizontal range.
DISPlay:ACPower:X[:SCALe]:AUTO	Rescales the horizontal axis automatically.
DISPlay:ACPower:X[:SCALe]:OFFSet	Sets or queries the minimum horizontal value (left edge).
DISPlay:ACPower:Y[:SCALe]	Sets or queries the vertical range.
DISPlay:ACPower:Y[:SCALe]:AUTO	Rescales the vertical axis automatically.
DISPlay:ACPower:Y[:SCALe]:OFFSet	Sets or queries the vertical offset.
DISPlay:ADEMod subgroup (Option 21 only)	**General purpose analog demodulation measurements**
DISPlay:ADEMod:MEASview:DELete	Deletes the measurement view.
DISPlay:ADEMod:MEASview:NEW	Displays a new measurement view.
DISPlay:ADEMod:MEASview:SELect	Sets or queries the measurement view.
DISPlay:{AM|FM|PM} subgroup (Option 21 only)	**AM/FM/PM measurement**
DISPlay:{AM|FM|PM}:MARKer:SHOW:STATe	Determines whether to show the readout for the selected marker.
DISPlay:{AM|FM|PM}:WINDow:TRACe:GRATicule:GRID:STATe	Determines whether to show the graticule grid on screen.
DISPlay:{AM|FM|PM}:X:RSCale	Rescales the horizontal axis automatically.
DISPlay:{AM|FM|PM}:X[:SCALe]:AUTO	Sets the horizontal scale automatically.
DISPlay:{AM|FM|PM}:X[:SCALe]:FULL	Sets or queries the horizontal scale.

Table 2-13: Display commands (cont.)

Command	Description
DISPlay:{AM|FM|PM}:X[:SCALe]:OFFSet	Sets or queries the minimum horizontal value (left edge).
DISPlay:{AM|FM|PM}:Y:RSCale	Rescales the vertical axis automatically.
DISPlay:{AM|FM|PM}:Y[:SCALe]	Sets or queries the vertical scale.
DISPlay:{AM|FM|PM}:Y[:SCALe]:OFFSet	Sets or queries the vertical offset.
DISPlay:AUDio subgroup	**Audio measurements**
DISPlay:AUDio:MEASview:DELete	Deletes the specified audio analysis view.
DISPlay:AUDio:MEASview:NEW	Displays a new audio analysis view.
DISPlay:AUDio:MEASview:SELect	Selects an audio analysis view on the screen.
DISPlay:AUDio:SPECtrum:FREQuency:[SCALe]:STARt	Sets or queries the start frequency (left edge) of the audio spectrum graph.
DISPlay:AUDio:SPECtrum:FREQuency:[SCALe]:STOP	Sets or queries the stop frequency (right edge) of the audio spectrum graph.
DISPlay:AUDio:SPECtrum:FREQuency:AUTO	Rescales the horizontal axis automatically to fit the waveform to the screen in the audio spectrum view.
DISPlay:AUDio:SPECtrum:MARKer:SHOW:STATe	Determines whether to show or hide the readouts for the selected marker in the audio spectrum view.
DISPlay:AUDio:SPECtrum:RESet:SCALe	Resets the horizontal and vertical scales in the audio spectrum view.
DISPlay:AUDio:SPECtrum:SCALe:LOG:STATe	Determines whether or not to set the horizontal axis logarithmic in the audio spectrum view.
DISPlay:AUDio:SPECtrum:SHOW:NHARmonic:THReshold	Sets or queries the appearance of the non-harmonic threshold in the audio spectrum view.
DISPlay:AUDio:SPECtrum:TABLe:SHOW:STATe	Determines whether to show or hide the results table in the audio spectrum view.
DISPlay:AUDio:SPECtrum:WINDow:TRACe:GRATicule:GRID:STATe	Determines whether to show or hide the graticule grid on the screen.
DISPlay:AUDio:SPECtrum:Y:[SCALe]	Sets or queries the vertical scale in the audio spectrum view.
DISPlay:AUDio:SPECtrum:Y:[SCALe]:AUTO	Rescales the vertical axis automatically in the audio spectrum view.
DISPlay:AUDio:SPECtrum:Y:[SCALe]:OFFSet	Sets or queries the vertical position (offset) in the audio spectrum view.
DISPlay:AVTime subgroup	**Amplitude versus Time measurement**
DISPlay:AVTime:LEGend:STATe	Shows or hides the trace legend in the amplitude versus time view.
DISPlay:AVTime:MARKer:SHOW:STATe	Determines whether to show the readout for the selected marker.
DISPlay:AVTime:RESet	Resets the horizontal and vertical scale to the default values.
DISPlay:AVTime:TRIGger:LEVel:STATe	Determines whether to show the power trigger level line on screen.
DISPlay:AVTime:WINDow:TRACe:GRATicule:GRID:STATe	Determines whether to show the graticule grid on screen.
DISPlay:AVTime:X:RSCale	Rescales the horizontal axis automatically.
DISPlay:AVTime:X[:SCALe]:AUTO	Sets the horizontal scale automatically.
DISPlay:AVTime:X[:SCALe]:AUTO:STATe	Determines whether to set the horizontal scale automatically or manually.
DISPlay:AVTime:X[:SCALe]:FULL	Sets or queries the horizontal scale.

Table 2-13: Display commands (cont.)

Command	Description
DISPlay:AVTime:X[:SCALe]:MAXimum?	Queries the upper limit of the horizontal scale setting range.
DISPlay:AVTime:X[:SCALe]:MINimum?	Queries the lower limit of the horizontal scale setting range.
DISPlay:AVTime:X[:SCALe]:OFFSet	Sets or queries the minimum horizontal value (left edge).
DISPlay:AVTime:X[:SCALe]:OFFSet:MAXimum?	Queries the upper limit of the horizontal offset setting range.
DISPlay:AVTime:X[:SCALe]:OFFSet:MINimum?	Queries the lower limit of the horizontal offset setting range.
DISPlay:AVTime:Y:RSCale	Rescales the vertical axis automatically.
DISPlay:AVTime:Y[:SCALe]:FULL	Sets or queries the vertical scale.
DISPlay:AVTime:Y[:SCALe]:OFFSet	Sets or queries the vertical offset.
DISPlay:CONSte subgroup (Option 21 only)	**Constellation measurement**
DISPlay:CONSte:MPHase	Sets or queries the phase multiplication constant for a CPM signal.
DISPlay:CONSte:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries whether to show the graticule grid on the screen.
DISPlay:DDEMod subgroup (Option 21 only)	**General purpose digital modulation measurements**
DISPlay:DDEMod:MEASview:DELete	Deletes the measurement view.
DISPlay:DDEMod:MEASview:NEW	Displays a new measurement view.
DISPlay:DDEMod:MEASview:SELect	Sets or queries the measurement view.
DISPlay:DDEMod:RADix	Sets or queries the base of symbols.
DISPlay:DDEMod:X[:SCALe]	Sets or queries the horizontal scale.
DISPlay:DDEMod:X[:SCALe]:AUTO	Sets the horizontal scale automatically.
DISPlay:DDEMod:X[:SCALe]:AUTO:STATe	Determines whether to set the horizontal scale automatically or manually.
DISPlay:DDEMod:X[:SCALe]:MAXimum?	Queries the upper limit of the horizontal scale setting range.
DISPlay:DDEMod:X[:SCALe]:MINimum?	Queries the lower limit of the horizontal scale setting range.
DISPlay:DDEMod:X[:SCALe]:OFFSet	Sets or queries the minimum horizontal value (left edge).
DISPlay:DDEMod:X[:SCALe]:OFFSet:MAXimum?	Queries the upper limit of the horizontal offset setting range.
DISPlay:DDEMod:X[:SCALe]:OFFSet:MINimum?	Queries the lower limit of the horizontal offset setting range.
DISPlay:DDEMod:X[:SCALe]:RESet	Presets the horizontal scale to the default value.
DISPlay:DIAGram subgroup (Option 21 only)	**Eye/Trellis diagram**
DISPlay:DIAGram:X[:SCALe]	Sets or queries the horizontal range.
DISPlay:DIAGram:X[:SCALe]:RESet	Presets the horizontal scale to the default value.
DISPlay:DIQVtime subgroup (Option 21 only)	**Demodulated I&Q versus Time measurement**
DISPlay:DIQVtime:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries whether to show the graticule grid on the screen.
DISPlay:DIQVtime:Y[:SCALe]	Sets or queries the vertical scale.
DISPlay:DIQVtime:Y[:SCALe]:AUTO	Sets the vertical scale automatically.
DISPlay:DIQVtime:Y[:SCALe]:OFFSet	Sets or queries the vertical offset (center point of the vertical axis).

Table 2-13: Display commands (cont.)

Command	Description
DISPlay:DPX subgroup	**DPXogram measurement**
DISPlay:DPX:DGRam:TIME[:SCALe]:OFFSet:DIVision	Sets or queries the DPXogram vertical time offset in divisions.
DISPlay:DPX:DGRam:TIME[:SCALe]:PDIVision	Sets or queries the DPXogram vertical time scale per division.
DISPlay:DPX:DGRam:TIME[:SCALe]:RESet	Presets the time scale to the default value for the DPXogram measurement.
DISPlay:DPX:DGRam:TSTamp:STATe	Determines whether to display the time stamp readout in the DPXogram display.
DISPlay:DPX:DGRam:Y[:SCALe]:AUTO	Rescales the height axis automatically to fit the waveform to the screen in the DPXogram display.
DISPlay:DPX:DGRam:Y[:SCALe]:RESet	Resets the height scale of the DPXogram display.
DISPlay:DPX:LEGend:STATe	Determines whether to show or hide the trace legend on the display.
DISPlay:DPX:PHASe:Y[:SCALe]:AXIS	Sets or queries the vertical axis representation.
DISPlay:DPX:PHASe:Y[:SCALe]:AXIS:OFFSet	Sets or queries the vertical offset in the DPX Phase view.
DISPlay:DPX:Y[:SCALe]:OFFSet	Sets or queries the vertical offset.
DISPlay:DPX:Y[:SCALe]:PDIVision	Sets or queries the vertical scale (per division).
DISPlay:DPX:WINDow:TRACe:GRATicule:GRID:STATe	Shows or hides the graticule grid on the screen
DISPlay:EDIagram subgroup (Option 21 only)	**Eye diagram**
DISPlay:EDIagram:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries whether to show the graticule grid on the screen.
DISPlay:EDIagram:Y[:SCALe]	Sets or queries the vertical scale.
DISPlay:EDIagram:Y[:SCALe]:AUTO	Sets the vertical scale automatically.
DISPlay:EDIagram:Y[:SCALe]:OFFSet	Sets or queries the vertical offset (center point of the vertical axis).
DISPlay:EVM subgroup (Option 21 only)	**EVM versus Time measurement**
DISPlay:EVM:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries the graticule grid view state for the EVM view.
DISPlay:EVM:Y[:SCALe]	Sets or queries the vertical scale.
DISPlay:EVM:Y[:SCALe]:AUTO	Sets the vertical scale automatically.
DISPlay:EVM:Y[:SCALe]:OFFSet	Sets or queries the minimum vertical value (bottom edge).
DISPlay:FDVTime subgroup (Option 21 only)	**Frequency deviation versus Time measurement**
DISPlay:FDVTime:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries whether to show the graticule grid on the Frequency deviation versus Time view.
DISPlay:FDVTime:Y[:SCALe]	Sets or queries the vertical range of the Frequency deviation versus Time graph.
DISPlay:FDVTime:Y[:SCALe]:AUTO	Sets the vertical scale automatically.
DISPlay:FDVTime:Y[:SCALe]:OFFSet	Sets or queries the vertical offset (center point of the vertical axis).
DISPlay:{FSETtling|PSETtling} subgroup	**Frequency and Phase Settling measurements**
DISPlay:{FSETtling|PSETtling}:MARKer:SHOW:STATe	Determines whether to show or hide the readout for the selected marker in the Frequency or Phase Settling views.
DISPlay:{FSETtling|PSETtling}:TIME:DECimal	Sets or queries the number of values to the right of the decimal point.

Table 2-13: Display commands (cont.)

Command	Description
DISPlay:{FSETtling|PSETtling}:WINDow:TRACe:GRATicule:GRID:STATe	Determines whether to show or hide the graticule grid on the screen.
DISPlay:{FSETtling|PSETtling}:X[:SCALe]	Sets or queries the horizontal scale (full-scale time) of the Settling Time graph.
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO	Sets the horizontal scale automatically to fit the waveform to the screen in the Frequency and Phase Settling views.
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO:STATe	Determines whether to set the horizontal scale automatically or manually.
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:MAXimum?	Queries the upper limit of the horizontal scale setting range.
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:MINimum?	Queries the lower limit of the horizontal scale setting range.
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet:MINimum?	Queries the lower limit of the horizontal offset setting range.
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet	Sets or queries the minimum horizontal value (left edge) of the Frequency versus Time graph.
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet:MAXimum?	Queries the upper limit of the horizontal offset setting range.
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet:MINimum?	Queries the lower limit of the horizontal offset setting range.
DISPlay:{FSETtling|PSETtling}:Y[:SCALe]	Sets or queries the vertical range of the Frequency versus Time graph.
DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:AUTO	Sets the vertical scale automatically to fit the waveform to the screen in the Frequency and Phase Settling views.
DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:OFFSet	Sets or queries the vertical offset (the value at the center of the vertical axis) in the Frequency and Phase Settling graphs.
DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:PDIVision	Sets or queries the vertical scale (per division) of the Frequency and Phase Settling graphs.
DISPlay:FVTime subgroup	**Frequency versus Time measurement**
DISPlay:FVTime:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries whether to show the graticule grid on the Frequency versus Time view.
DISPlay:FVTime:X[:SCALe]	Sets or queries the horizontal scale.
DISPlay:FVTime:X[:SCALe]:AUTO	Sets the horizontal scale automatically.
DISPlay:FVTime:X[:SCALe]:AUTO:STATe	Determines whether to set the horizontal scale automatically or manually.
DISPlay:FVTime:X[:SCALe]:MAXimum?	Queries the upper limit of the horizontal scale setting range.
DISPlay:FVTime:X[:SCALe]:MINimum?	Queries the lower limit of the horizontal scale setting range.
DISPlay:FVTime:X[:SCALe]:OFFSet	Sets or queries the minimum horizontal value (left edge).
DISPlay:FVTime:X[:SCALe]:OFFSet:MAXimum?	Queries the upper limit of the horizontal offset setting range.
DISPlay:FVTime:X[:SCALe]:OFFSet:MINimum?	Queries the lower limit of the horizontal offset setting range.
DISPlay:FVTime:Y[:SCALe]	Sets or queries the vertical scale.
DISPlay:FVTime:Y[:SCALe]:AUTO	Sets the vertical scale automatically.
DISPlay:FVTime:Y[:SCALe]:OFFSet	Sets or queries the vertical offset.

Table 2-13: Display commands (cont.)

Command	Description
DISPlay:GENeral subgroup	**General signal viewing**
DISPlay:GENeral:MEASview:DELete	Deletes the measurement view.
DISPlay:GENeral:MEASview:NEW	Displays a new measurement view.
DISPlay:GENeral:MEASview:SELect	Sets or queries the measurement view.
DISPlay:GPRF subgroup	**General purpose RF measurements**
DISPlay:GPRF:MEASview:DELete	Deletes the measurement view.
DISPlay:GPRF:MEASview:NEW	Displays a new measurement view.
DISPlay:GPRF:MEASview:SELect	Sets or queries the measurement view.
DISPlay:IQVTime subgroup	**RF I&Q versus Time measurement**
DISPlay:IQVTime:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries whether to show the graticule grid on the screen.
DISPlay:IQVTime:X[:SCALe]	Sets or queries the horizontal scale.
DISPlay:IQVTime:X[:SCALe]:AUTO	Sets the horizontal scale automatically.
DISPlay:IQVTime:X[:SCALe]:AUTO:STATe	Determines whether to set the horizontal scale automatically or manually.
DISPlay:IQVTime:X[:SCALe]:MAXimum?	Queries the upper limit of the horizontal scale setting range.
DISPlay:IQVTime:X[:SCALe]:MINimum?	Queries the lower limit of the horizontal scale setting range.
DISPlay:IQVTime:X[:SCALe]:OFFSet	Sets or queries the minimum horizontal value (left edge).
DISPlay:IQVTime:X[:SCALe]:OFFSet:MAXimum?	Queries the upper limit of the horizontal offset setting range.
DISPlay:IQVTime:X[:SCALe]:OFFSet:MINimum?	Queries the lower limit of the horizontal offset setting range.
DISPlay:IQVTime:Y[:SCALe]	Sets or queries the vertical scale.
DISPlay:IQVTime:Y[:SCALe]:AUTO	Sets the vertical scale automatically.
DISPlay:IQVTime:Y[:SCALe]:OFFSet	Sets or queries the vertical offset.
DISPlay:IQVTime:Y[:SCALe]:RESCale	Rescales the vertical scale.
DISPlay:MCPower subgroup	**MCPR measurement**
DISPlay:MCPower:MARKer:SHOW:STATe	Determines whether to show the readout for the selected marker.
DISPlay:MCPower:PLEVel:SHOW:STATe	Determines whether to show the power levels.
DISPlay:MCPower:RESet:SCALe	Resets the horizontal and vertical scale to the default values.
DISPlay:MCPower:WINDow:TRACe:GRATicule:GRID:STATe	Determines whether to show the graticule grid on screen.
DISPlay:MCPower:X[:SCALe]	Sets or queries the horizontal range.
DISPlay:MCPower:X[:SCALe]:AUTO	Rescales the horizontal axis automatically.
DISPlay:MCPower:X[:SCALe]:OFFSet	Sets or queries the minimum horizontal value (left edge).
DISPlay:MCPower:Y[:SCALe]	Sets or queries the vertical range.
DISPlay:MCPower:Y[:SCALe]:AUTO	Rescales the vertical axis automatically.
DISPlay:MCPower:Y[:SCALe]:OFFSet	Sets or queries the vertical offset.

Table 2-13: Display commands (cont.)

Command	Description
DISPlay:MERRor subgroup (Option 21 only)	**Magnitude error versus Time measurement**
DISPlay:MERRor:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries the graticule grid view state for the Magnitude error versus Time view.
DISPlay:MERRor:Y[:SCALe]	Sets or queries the vertical scale.
DISPlay:MERRor:Y[:SCALe]:AUTO	Sets the vertical scale automatically.
DISPlay:MERRor:Y[:SCALe]:OFFSet	Sets or queries the minimum vertical value (bottom edge).
DISPlay:OBWidth subgroup	**Occupied Bandwidth measurement**
DISPlay:OBWidth:MARKer:SHOW:STATe	Determines whether to show the readout for the selected marker.
DISPlay:OBWidth:RESet:SCALe	Resets the horizontal and vertical scale to the default values.
DISPlay:OBWidth:SELected:BANDwidth	Sets or queries the bandwidth (OBW or x dB BW) to measure.
DISPlay:OBWidth:WINDow:TRACe:GRATicule:GRID:STATe	Determines whether to show the graticule grid on screen.
DISPlay:OBWidth:X[:SCALe]	Sets or queries the horizontal range.
DISPlay:OBWidth:X[:SCALe]:AUTO	Rescales the horizontal axis automatically.
DISPlay:OBWidth:X[:SCALe]:OFFSet	Sets or queries the minimum horizontal value (left edge).
DISPlay:OBWidth:Y[:SCALe]	Sets or queries the vertical range.
DISPlay:OBWidth:Y[:SCALe]:AUTO	Rescales the vertical axis automatically.
DISPlay:OBWidth:Y[:SCALe]:OFFSet	Sets or queries the vertical offset.
DISPlay:OFDM subgroup	**OFDM measurements**
DISPlay:OFDM:CONSte[:SCALe]	Sets or queries the horizontal offset.
DISPlay:OFDM:CONSte:X:OFFSet	Sets or queries the vertical offset.
DISPlay:OFDM:CONSte:Y:OFFSet	Sets or queries the scale.
DISPlay:OFDM:CRESponse:FREQuency:AUTO	Rescales the horizontal axis automatically.
DISPlay:OFDM:CRESponse:FREQuency:OFFSet	Sets or queries the frequency offset.
DISPlay:OFDM:CRESponse:FREQuency[:SCALe]	Sets or queries the horizontal range.
DISPlay:OFDM:CRESponse:MAGNitude:AUTO	Rescales the magnitude automatically.
DISPlay:OFDM:CRESponse:MAGNitude:OFFSet	Sets or queries the magnitude offset.
DISPlay:OFDM:CRESponse:MAGNitude:PDIVision	Sets or queries the scale (per division) of the magnitude.
DISPlay:OFDM:CRESponse:PHASe:AUTO	Rescales the vertical axis automatically.
DISPlay:OFDM:CRESponse:PHASe:OFFSet	Sets or queries the phase offset.
DISPlay:OFDM:CRESponse:PHASe[:SCALe]	Sets the vertical scale automatically.
DISPlay:OFDM:CRESponse:WINDow:SELect:PLOT	Sets or queries the plot.
DISPlay:OFDM:CRESponse:WINDow:TRACe:GRATicule:GRID:STATe	Determines whether to show or hide the graticule.
DISPlay:OFDM:EVM:FREQuency[:SCALe]	Sets or queries the horizontal range.
DISPlay:OFDM:EVM:FREQuency:AUTO	Rescales the frequency automatically to fit the waveform to the screen.
DISPlay:OFDM:EVM:FREQuency:OFFSet	Sets or queries the frequency offset.

Table 2-13: Display commands (cont.)

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

Table 2-13: Display commands (cont.)

Command	Description
DISPlay:OFDM:PERRor:MARKer:SHOW:STATe	Determines whether to show or hide the readout for the selected marker.
DISPlay:OFDM:PERRor:TIME[:SCALe]	Sets or queries the time scale.
DISPlay:OFDM:PERRor:TIME[:SCALe]:AUTO	Rescales the time automatically to fit the waveform to the screen.
DISPlay:OFDM:PERRor:TIME[:SCALe]:OFFSet	Sets or queries the time offset.
DISPlay:OFDM:PERRor:WINDow:SELect:PLOT	Sets or queries the plot.
DISPlay:OFDM:PERRor:WINDow:TRACe:GRATicule:GRID:STATe	Determines whether to show or hide the graticule.
DISPlay:OFDM:PERRor:Y[:SCALe]	Sets or queries the vertical range.
DISPlay:OFDM:PERRor:Y:AUTO	Rescales the vertical scale automatically to fit the waveform.
DISPlay:OFDM:PERRor:Y:OFFSet	Sets or queries the vertical offset.
DISPlay:OFDM:POWer:FREQuency[:AUTO]	Rescales the frequency automatically to fit the waveform to the screen.
DISPlay:OFDM:POWer:FREQuency[:OFFSet]	Sets or queries the frequency offset.
DISPlay:OFDM:POWer:FREQuency[:SCALe]	Sets or queries the horizontal range.
DISPlay:OFDM:POWer:MARKer:SHOW:STATe	Determines whether to show or hide the readout for the selected marker.
DISPlay:OFDM:POWer:TIME[:AUTO	Rescales the time automatically to fit the waveform to the screen.
DISPlay:OFDM:POWer:TIME:OFFSet	Sets or queries the time offset.
DISPlay:OFDM:POWer:TIME[:SCALe]	Sets or queries the time scale.
DISPlay:OFDM:POWer:WINDow:SELect:PLOT	Sets or queries the plot.
DISPlay:OFDM:POWer:WINDow:TRACe:GRATicule:GRID:STATe	Determines whether to show or hide the graticule.
DISPlay:OFDM:POWer:Y:AUTO	Rescales the vertical scale automatically to fit the waveform.
DISPlay:OFDM:POWer:Y:OFFSet	Sets or queries the vertical offset.
DISPlay:OFDM:POWer:Y[:SCALe]	Sets or queries the vertical range.
DISPlay:PERRor subgroup (Option 21 only)	**Phase error versus Time measurement**
DISPlay:PERRor:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries the graticule grid view state for the Phase error versus Time view.
DISPlay:PERRor:Y[:SCALe]	Sets or queries the vertical scale.
DISPlay:PERRor:Y[:SCALe]:AUTO	Sets the vertical scale automatically.
DISPlay:PERRor:Y[:SCALe]:OFFSet	Sets or queries the minimum vertical value (bottom edge).
DISPlay:PHVTime subgroup	**Phase versus Time measurement**
DISPlay:PHVTime:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries whether to show the graticule grid on the screen.
DISPlay:PHVTime:X[:SCALe]	Sets or queries the horizontal scale.
DISPlay:PHVTime:X[:SCALe]:AUTO	Sets the horizontal scale automatically.
DISPlay:PHVTime:X[:SCALe]:AUTO:STATe	Determines whether to set the horizontal scale automatically or manually.
DISPlay:PHVTime:X[:SCALe]:MAXimum?	Queries the upper limit of the horizontal scale setting range.
DISPlay:PHVTime:X[:SCALe]:MINimum?	Queries the lower limit of the horizontal scale setting range.
DISPlay:PHVTime:X[:SCALe]:OFFSet	Sets or queries the minimum horizontal value (left edge).

Table 2-13: Display commands (cont.)

Command	Description
DISPlay:PHVTime:X[:SCALe]:OFFSet:MAXimum?	Queries the upper limit of the horizontal offset setting range.
DISPlay:PHVTime:X[:SCALe]:OFFSet:MINimum?	Queries the lower limit of the horizontal offset setting range.
DISPlay:PHVTime:Y[:SCALe]	Sets or queries the vertical scale.
DISPlay:PHVTime:Y[:SCALe]:AUTO	Sets the vertical scale automatically.
DISPlay:PHVTime:Y[:SCALe]:AXIS	Sets or queries the vertical axis representation.
DISPlay:PHVTime:Y[:SCALe]:AXIS:REFerence	Sets or queries the reference time for phase.
DISPlay:PHVTime:Y[:SCALe]:OFFSet	Sets or queries the vertical offset.
DISPlay:PHVTime:Y[:SCALe]:RESCale	Rescales the vertical scale.
DISPlay:PNOise subgroup (Option 11 only)	**Phase noise measurements**
DISPlay:PNOise:LEGend:STATe	Sets or queries showing the trace legend on the display.
DISPlay:PNOise:MARKer:SHOW:STATe	Determines whether to show the readout for the selected marker.
DISPlay:PNOise:RESet:SCALe	Resets the horizontal and vertical scale to the default values.
DISPlay:PNOise:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries whether to show the graticule grid on the screen.
DISPlay:PNOise:X[:SCALe]:AUTO	Rescales the horizontal axis automatically.
DISPlay:PNOise:X[:SCALe]:STARt	Sets or queries the start frequency of the graph.
DISPlay:PNOise:X[:SCALe]:STOP	Sets or queries the stop frequency of the graph.
DISPlay:PNOise:Y[:SCALe]	Sets or queries the vertical scale.
DISPlay:PNOise:Y[:SCALe]:AUTO	Rescales the vertical axis automatically.
DISPlay:PNOise:Y[:SCALe]:OFFSet	Sets or queries the vertical offset.
DISPlay:PNOise:Y[:SCALe]:PDIVision	Sets or queries the vertical scale (per division).
DISPlay:PULSe subgroup	**Pulsed RF measurements**
DISPlay:PULSe:MEASview:DELete	Deletes the measurement view.
DISPlay:PULSe:MEASview:NEW	Displays a new measurement view.
DISPlay:PULSe:MEASview:SELect	Sets or queries the measurement view.
DISPlay:PULSe:RESult:ATX	Sets or queries whether to show the average transmitted power result.
DISPlay:PULSe:RESult:AVERage	Sets or queries whether to show the average on power result.
DISPlay:PULSe:RESult:DRODb	Sets or queries whether to show the droop in the results table.
DISPlay:PULSe:RESult:DROop	Sets or queries showing the droop measurement result in the pulse table
DISPlay:PULSe:RESult:DUTPct	Sets or queries whether to show the duty factor (%) result.
DISPlay:PULSe:RESult:DUTRatio	Sets or queries whether to show the duty factor (ratio) result.
DISPlay:PULSe:RESult:FALL	Sets or queries whether to show the fall time in the results table.
DISPlay:PULSe:RESult:FDELta	Sets or queries whether to show the delta frequency result.
DISPlay:PULSe:RESult:FRDeviation	Sets or queries whether to show the frequency deviation result.
DISPlay:PULSe:RESult:IRAMplitude	Sets or queries showing the Impulse Response Amplitude measurement result in the pulse table.

Table 2-13: Display commands (cont.)

Command	Description
DISPlay:PULSe:RESult:IRTime	Sets or queries showing the Impulse Response Time measurement result in the pulse table.
DISPlay:PULSe:RESult:MFReqerror	Sets or queries whether to show the maximum frequency error result.
DISPlay:PULSe:RESult:MPHerror	Sets or queries whether to show the maximum phase error result.
DISPlay:PULSe:RESult:OVEDb	Sets or queries whether to show the Overshoot measurement result in dB in the pulse table.
DISPlay:PULSe:RESult:OVERshoot	Sets or queries whether to show the Overshoot measurement result in the pulse table.
DISPlay:PULSe:RESult:PHDeviation	Sets or queries whether to show the phase deviation result.
DISPlay:PULSe:RESult:PPFRequency	Sets or queries whether to show the pulse-pulse frequency result.
DISPlay:PULSe:RESult:PPOWer	Sets or queries whether to show the peak power in the results table.
DISPlay:PULSe:RESult:PPPHase	Sets or queries whether to show the pulse-pulse carrier phase result.
DISPlay:PULSe:RESult:RINTerval	Sets or queries whether to show the repetition interval result.
DISPlay:PULSe:RESult:RIPDb	Sets or queries showing the ripple measurement result in dB in the pulse table.
DISPlay:PULSe:RESult:RIPPle	Sets or queries showing the ripple measurement result in the pulse table.
DISPlay:PULSe:RESult:RISE	Sets or queries whether to show the rise time in the results table.
DISPlay:PULSe:RESult:RMSFreqerror	Sets or queries whether to show the RMS frequency error result.
DISPlay:PULSe:RESult:RMSPherror	Sets or queries whether to show the RMS phase error result.
DISPlay:PULSe:RESult:RRATe	Sets or queries whether to show the repetition rate result.
DISPlay:PULSe:RESult:TIME	Sets or queries whether to show the time in the results table.
DISPlay:PULSe:RESult:WIDTh	Sets or queries whether to show the pulse width in the results table.
DISPlay:PULSe:SELect:NUMBer	Sets or queries the pulse to measure.
DISPlay:PULSe:SELect:RESult	Sets or queries which result is shown in the trace and statistics views.
DISPlay:PULSe:STATistics:MARKer:SHOW:STATe	Sets or queries whether to show the marker readout in the graph.
DISPlay:PULSe:STATistics:PLOT	Sets or queries how to show the statistics graph.
DISPlay:PULSe:STATistics:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries whether to show the graticule grid in the statistics view.
DISPlay:PULSe:STATistics:X:RSCale	Rescales the horizontal axis of the statistics graph.
DISPlay:PULSe:STATistics:X[:SCALe]:NUMBer	Sets or queries the horizontal scale (the number of pulses per division).
DISPlay:PULSe:STATistics:X[:SCALe]:OFFSet	Sets or queries the minimum horizontal value in the statistics view.
DISPlay:PULSe:STATistics:Y:RSCale	Rescales the vertical axis of the statistics graph.
DISPlay:PULSe:STATistics:Y[:SCALe]:FULL	Sets or queries the vertical full-scale in the statistics view.
DISPlay:PULSe:STATistics:Y[:SCALe]:OFFSet	Sets or queries the vertical offset in the statistics view.
DISPlay:PULSe:STATistics:Y[:SCALe]:STOP?	Queries the minimum vertical value in the statistics view.
DISPlay:PULSe:TRACe:MARKer:SHOW:STATe	Sets or queries whether to show the marker readout in the trace view.
DISPlay:PULSe:TRACe:POINt:SHOW	Sets or queries whether to show the measurement points and lines.

Table 2-13: Display commands (cont.)

Command	Description
DISPlay:PULSe:TRACe:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries whether to show the graticule grid in the trace view.
DISPlay:PULSe:TRACe:X:RSCale	Rescales the horizontal axis of the pulse trace view.
DISPlay:PULSe:TRACe:X[:SCALe]	Sets or queries the horizontal full scale in the pulse trace view.
DISPlay:PULSe:TRACe:X[:SCALe]:FULL	Sets or queries the full-scale reference for the horizontal rescale.
DISPlay:PULSe:TRACe:X[:SCALe]:OFFSet	Sets or queries the minimum horizontal value in the pulse trace view.
DISPlay:PULSe:TRACe:X[:SCALe]:PDIVision	Sets or queries the horizontal full scale in the pulse trace view.
DISPlay:PULSe:TRACe:Y:RSCale	Rescales the vertical axis of the pulse trace view.
DISPlay:PULSe:TRACe:Y[:SCALe]:FULL	Sets or queries the vertical full scale in the pulse trace view.
DISPlay:PULSe:TRACe:Y[:SCALe]:OFFSet	Sets or queries the vertical offset in the pulse trace view.
DISPlay:PULSe:TRACe:Y[:SCALe]:STOP?	Queries the minimum vertical value in the pulse trace view.
DISPlay:SEM subgroup	**Spectral Emissions Mask measurements**
DISPlay:SEM:MARKer:SHOW:STATe	Determines whether to show or hide the readout for the selected marker.
DISPlay:SEM:RESet:SCALe	Resets the scale.
DISPlay:SEM:SHOW:LIMit	Sets or queries the appearance of the limits.
DISPlay:SEM:WINDow:TRACe:GRATicule:GRID:STATe	Determines whether to show or hide the graticule.
DISPlay:SEM:X[:SCALe]:AUTO	Rescales the horizontal scale automatically to fit the waveform.
DISPlay:SEM:X[:SCALe]:START	Sets or queries the start frequency.
DISPlay:SEM:X[:SCALe]:STOP	Sets or queries the stop frequency.
DISPlay:SEM:Y[:SCALe]	Sets or queries the vertical range.
DISPlay:SEM:Y[:SCALe]:AUTO	Rescales the vertical scale automatically to fit the waveform.
DISPlay:SEM:Y[:SCALe]:OFFSet	Sets or queries the vertical offset.
DISPlay:SGRam subgroup	**Spectrogram measurement**
DISPlay:SGRam:FREQuency:AUTO	Rescales the horizontal (frequency) axis automatically.
DISPlay:SGRam:FREQuency:OFFSet	Sets or queries the horizontal (frequency) offset.
DISPlay:SGRam:FREQuency:SCAle	Sets or queries the horizontal (frequency) range.
DISPlay:SGRAM:MARKer:SHOW:STATe	Sets or queries the marker readout.
DISPlay:SGRAM:SELected:TIMestamp	Sets or queries the time stamp readout.
DISPlay:SGRam:TIME:AUTO	Rescales the vertical axis automatically.
DISPlay:SGRam:TIME:OFFSet	Sets or queries the vertical axis (time) offset (bottom line number).
DISPlay:SGRam:TIME:OFFSet:DIVisions	Sets or queries the Vertical offset or Position in divisions.
DISPlay:SGRam:TIME:OVERlap:PERCent	Sets or queries the Overlap in percent.
DISPlay:SGRam:TIME:SCAle	Sets or queries the vertical scale (the amount of time in each line).
DISPlay:SGRam:TIME:SCALe:PER:DIVision	Sets or queries the vertical Time/division value.
DISPlay:SGRam:TIME:SPECtrums:PERLine?	Queries the Spectrums/Line value.

Table 2-13: Display commands (cont.)

Command	Description
DISPlay:SPECtrum subgroup	**Spectrum measurement**
DISPlay:SPECtrum:FREQuency:AUTO	Rescales the horizontal (frequency) axis automatically.
DISPlay:SPECtrum:FREQuency:OFFSet	Sets or queries the horizontal (frequency) offset.
DISPlay:SPECtrum:FREQuency[:SCAle]	Sets or queries the horizontal (frequency) range.
DISPlay:SPECtrum:MARKer:NOISe:MODE	Sets or queries whether to enable the marker noise mode.
DISPlay:SPECtrum:SCALe:LOG:STATe	Sets or queries whether to set the horizontal axis logarithmic.
DISPlay:SPECtrum:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries whether to show the graticule grid.
DISPlay:SPECtrum:WINDow:TRACe:LEGend:STATe	Show or hide the trace legend in the Spectrum view.
DISPlay:SPECtrum:X:LABel	Sets or queries the labels for the horizontal axis.
DISPlay:SPECtrum:Y[:SCALe]	Sets or queries the vertical range.
DISPlay:SPECtrum:Y[:SCALe]:AUTO	Rescales the vertical axis automatically.
DISPlay:SPECtrum:Y[:SCALe]:OFFSet	Sets or queries the vertical offset.
DISPlay:SPECtrum:Y[:SCALe]:PDIVision	Sets or queries the vertical scale (per division).
DISPlay:SPECtrum:Y[:SCALe]:RESet	Resets the vertical scale to the default values.
DISPlay:SPURious subgroup	**Spurious measurement**
DISPlay:SPURious:MARKer:SHOW:STATe	Sets or queries whether to show the readout for the selected marker.
DISPlay:SPURious:RESet:SCALe	Resets the horizontal and vertical scale to the default values.
DISPlay:SPURious:SCALe:LOG:STATe	Sets or queries whether to set the horizontal axis logarithmic.
DISPlay:SPURious:SELect:NUMBer	Sets or queries the spurious number.
DISPlay:SPURious:SHOW:LIMit	Sets or queries how to display the limits.
DISPlay:SPURious:WINDow:TRACe:GRATicule:GRID:STATe	Determines whether to show the graticule grid on screen.
DISPlay:SPURious:X[:SCALe]:AUTO	Rescales the horizontal axis automatically.
DISPlay:SPURious:X[:SCALe]:STARt	Sets or queries the minimum horizontal value of the spectrum graph.
DISPlay:SPURious:X[:SCALe]:STOP	Sets or queries the maximum horizontal value of the spectrum graph.
DISPlay:SPURious:Y[:SCALe]	Sets or queries the vertical range of the spectrum graph.
DISPlay:SPURious:Y[:SCALe]:AUTO	Rescales the vertical axis automatically.
DISPlay:SPURious:Y[:SCALe]:OFFSet	Sets or queries the vertical offset of the spectrum graph.
DISPlay:TDIagram subgroup (Option 21 only)	**Trellis diagram**
DISPlay:TDIagram:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries whether to show the graticule grid on the screen.
DISPlay:TDIagram:Y[:SCALe]	Sets or queries the vertical scale.
DISPlay:TDIagram:Y[:SCALe]:AUTO	Sets the vertical scale automatically.
DISPlay:TDIagram:Y[:SCALe]:OFFSet	Sets or queries the vertical offset (center point of the vertical axis).
DISPlay:TOVerview subgroup	**Time overview**
DISPlay:TOVerview:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries whether to show the graticule grid in the time overview.

Table 2-13: Display commands (cont.)

Command	Description
DISPlay:TOVerview:X[:SCALe]	Sets or queries the horizontal scale.
DISPlay:TOVerview:X[:SCALe]:AUTO	Sets the horizontal scale and offset automatically.
DISPlay:TOVerview:X[:SCALe]:OFFSet	Sets or queries the minimum horizontal value (left edge).
DISPlay:TOVerview:Y[:SCALe]	Sets or queries the vertical scale.
DISPlay:TOVerview:Y[:SCALe]:AUTO	Sets the vertical scale and offset automatically.
DISPlay:TOVerview:Y[:SCALe]:OFFSet	Sets or queries the vertical offset.
DISPlay:TOVerview:Y[:SCALe]:RESCale	Rescales the vertical scale.
DISPlay:WINDow subgroup	**Window display**
DISPlay:WINDow:ACTive:MEASurement?	Queries the active measurement views
DISPlay:WINDow:COLor:SCHeme	Sets or queries the color scheme for displaying traces and background on the screen.
DISPlay:WINDow:OPTimized:MEASurement?	Queries the measurement views that are optimized.
DISPlay:WLAN subgroup	**Wireless LAN measurements**
DISPlay:WLAN:CONSte[:SCALe]	Sets the magnification value (zoom factor).
DISPlay:WLAN:CONSte:X:OFFSet	Sets or queries the value of the offset of the horizontal axis.
DISPlay:WLAN:CONSte:Y:OFFSet	Sets or queries the value of the offset of the vertical axis.
DISPlay:WLAN:CRESponse:AUTO	Automatically rescales the horizontal and vertical values.
DISPlay:WLAN:CRESponse:FREQuency:AUTO	Automatically rescales the horizontal subcarrier (frequency) start and stop values.
DISPlay:WLAN:CRESponse:FREQuency:STARt	Specifies or queries the horizontal start value.
DISPlay:WLAN:CRESponse:FREQuency:STOP	Specifies or queries the horizontal stop value.
DISPlay:WLAN:CRESponse:MAGNitude:AUTO	Rescales the vertical magnitude offset (position) and scale values.
DISPlay:WLAN:CRESponse:MAGNitude:OFFSet	Sets or queries the vertical offset.
DISPlay:WLAN:CRESponse:MAGNitude[:SCALe]	Sets or queries the value of the vertical scale.
DISPlay:WLAN:CRESponse:MARKer:SHOW:STATe	Shows or hides the marker readout.
DISPlay:WLAN:CRESponse:PHASe:AUTO	Rescales the vertical phase value automatically to fit the waveform to the screen.
DISPlay:WLAN:CRESponse:PHASe:OFFSet	Sets or queries the vertical offset.
DISPlay:WLAN:CRESponse:PHASe[:SCALe]	Sets or queries the phase value.
DISPlay:WLAN:CRESponse:WINDow:SELect:PLOT	Specifies or queries which graphs are displayed.
DISPlay:WLAN:CRESponse:WINDow:TRACe:GRATicule:GRID:STATe	Sets or queries whether to show the graticule grid.
DISPlay:WLAN:EVM:FREQuency:AUTO	Automatically rescales the horizontal subcarrier (frequency) start and stop values.
DISPlay:WLAN:EVM:FREQuency:RESet	Automatically resets the horizontal subcarrier (frequency) start and stop values.
DISPlay:WLAN:EVM:FREQuency:STARt	Specifies or queries the horizontal start value.
DISPlay:WLAN:EVM:FREQuency:STOP	Specifies or queries the horizontal stop value.

Table 2-13: Display commands (cont.)

Command	Description
DISPlay:WLAN:EVM:MARKer:SHOW:STATe	Shows or hides the Marker readout.
DISPlay:WLAN:EVM[:SCALe]:AUTO	Automatically adjusts all vertical and horizontal values.
DISPlay:WLAN:EVM:TIME:AUTO	Sets the horizontal start and stop values to fit the waveform.
DISPlay:WLAN:EVM:TIME:AUTO:STATe	Automatically sets the horizontal symbol scale value.
DISPlay:WLAN:EVM:TIME:RESet	Resets the horizontal symbol start and stop values.
DISPlay:WLAN:EVM:TIME:STARt	Sets or queries the horizontal symbol start value.
DISPlay:WLAN:EVM:TIME:STOP	Sets or queries the horizontal symbol stop value
DISPlay:WLAN:EVM:WINDow:SELect:PLOT	Specifies or queries which graphs are displayed in the WLAN EVM display.
DISPlay:WLAN:EVM:WINDow:TRACe:GRATicule:GRID:STATe	Shows or hides the graticule grid on the screen.
DISPlay:WLAN:EVM:Y:AUTO	Automatically sets the vertical scale and position values.
DISPlay:WLAN:EVM:Y:OFFSet	Sets or queries the value of the offset from the center.
DISPlay:WLAN:EVM:Y:RESet	Resets the vertical scale and position (offset) values.
DISPlay:WLAN:EVM:Y[:SCALe]	Sets or queries the vertical scale value.
DISPlay:WLAN:FLATness:AUTO	Automatically rescales the horizontal and vertical axes.
DISPlay:WLAN:FLATness:X:AUTO	Automatically rescales the horizontal axis.
DISPlay:WLAN:FLATness:X:OFFSet	Sets or queries the value of the offset of the horizontal axis from the center.
DISPlay:WLAN:FLATness:X[:SCALe]	Sets or queries the horizontal scale value.
DISPlay:WLAN:FLATness:Y:AUTO	Automatically rescales the vertical axis scale and position values.
DISPlay:WLAN:FLATness:Y:OFFSet	Sets or queries the value of the offset from the center.
DISPlay:WLAN:FLATness:Y[:SCALe]	Sets or queries the vertical scale value.
DISPlay:WLAN:MEASview:DELete	Deletes the selected measurement view.
DISPlay:WLAN:MEASview:NEW	Displays a new measurement view.
DISPlay:WLAN:MEASview:SELect	Selects a WLAN measurement view.
DISPlay:WLAN:MERRor:FREQuency:AUTO	Automatically rescales the horizontal subcarrier (frequency) start and stop values.
DISPlay:WLAN:MERRor:FREQuency:RESet	Automatically resets the horizontal subcarrier (frequency) start and stop values.
DISPlay:WLAN:MERRor:FREQuency:STARt	Specifies or queries the horizontal start value.
DISPlay:WLAN:MERRor:FREQuency:STOP	Specifies or queries the horizontal stop value.
DISPlay:WLAN:MERRor:MARKer:SHOW:STATe	Shows or hides the marker readout.
DISPlay:WLAN:MERRor[:SCALe]:AUTO	Automatically adjusts all vertical and horizontal values.
DISPlay:WLAN:MERRor:TIME:AUTO	Sets the horizontal start and stop values.
DISPlay:WLAN:MERRor:TIME:AUTO:STATe	Automatically sets the horizontal symbol scale value.
DISPlay:WLAN:MERRor:TIME:RESet	Resets the horizontal symbol start and stop values.
DISPlay:WLAN:MERRor:TIME:STARt	Sets or queries the horizontal symbol start value.
DISPlay:WLAN:MERRor:TIME:STOP	Sets or queries the horizontal symbol stop value.

Table 2-13: Display commands (cont.)

Command	Description
DISPlay:WLAN:MERRor:WINDow:SELect:PLOT	Specifies or queries which graphs are displayed.
DISPlay:WLAN:MERRor:WINDow:TRACe:GRATicule:GRID:STATe	Shows or hides the graticule grid on the display.
DISPlay:WLAN:MERRor:Y:AUTO	Automatically sets the vertical scale and position values.
DISPlay:WLAN:MERRor:Y:OFFSet	Sets or queries the offset from the center (vertical position).
DISPlay:WLAN:MERRor:Y:RESet	Resets the vertical scale and position (offset).
DISPlay:WLAN:MERRor:Y[:SCALe]	Sets or queries the vertical scale value.
DISPlay:WLAN:PERRor:FREQuency:AUTO	Automatically rescales the horizontal subcarrier (frequency) start and stop values.
DISPlay:WLAN:PERRor:FREQuency:RESet	Automatically resets the horizontal subcarrier (frequency) start and stop values.
DISPlay:WLAN:PERRor:FREQuency:STARt	Specifies or queries the horizontal start value.
DISPlay:WLAN:PERRor:FREQuency:STOP	Specifies or queries the horizontal stop value.
DISPlay:WLAN:PERRor:MARKer:SHOW:STATe	Show or hides the Marker readout.
DISPlay:WLAN:PERRor[:SCALe]:AUTO	Automatically adjusts all vertical and horizontal settings.
DISPlay:WLAN:PERRor:TIME:AUTO	Automatically sets the horizontal symbol start and stop values.
DISPlay:WLAN:PERRor:TIME:AUTO:STATe	Automatically sets the horizontal symbol scale value.
DISPlay:WLAN:PERRor:TIME:RESet	Resets the horizontal symbol start and stop values.
DISPlay:WLAN:PERRor:TIME:STARt	Sets or queries the horizontal symbol start value.
DISPlay:WLAN:PERRor:TIME:STOP	Sets or queries the horizontal symbol stop value.
DISPlay:WLAN:PERRor:WINDow:SELect:PLOT	Specifies or queries which displays are displayed.
DISPlay:WLAN:PERRor:WINDow:TRACe:GRATicule:GRID:STATe	Shows or hides the graticule grid on the screen.
DISPlay:WLAN:PERRor:Y:AUTO	Automatically selects the vertical scale and position values.
DISPlay:WLAN:PERRor:Y:OFFSet	Sets or queries the value of the offset from the center (vertical position).
DISPlay:WLAN:PERRor:Y:RESet	Resets the vertical scale and position (offset) values.
DISPlay:WLAN:PERRor:Y[:SCALe]	Sets or queries the vertical scale value.
DISPlay:WLAN:PVTime:BURSt:X[:SCALe]	Sets or queries the value of the scale (width).
DISPlay:WLAN:PVTime:BURSt:X[:SCALe]:AUTO	Automatically sets the starting time (position) and scale (width) values
DISPlay:WLAN:PVTime:BURSt:X[:SCALe]:OFFSet	Sets or queries the starting time (position) value.
DISPlay:WLAN:PVTime:BURSt:X[:SCALe]:RESet	Resets the starting time (position) and scale (width) values.
DISPlay:WLAN:PVTime:FALL:X[:SCALe]	Sets or queries the scale (width) value.
DISPlay:WLAN:PVTime:FALL:X[:SCALe]:AUTO	Automatically sets the starting time (position) and scale (width) values.
DISPlay:WLAN:PVTime:FALL:X[:SCALe]:OFFSet	Sets or queries the value for the starting time (offset).
DISPlay:WLAN:PVTime:FALL:X[:SCALe]:RESet	Resets the starting time (position) and scale (width) values.
DISPlay:WLAN:PVTime:MARKer:SHOW:STATe	Shows or hides the marker readout.
DISPlay:WLAN:PVTime:RISE:X[:SCALe]	Sets or queries the scale (width) value.
DISPlay:WLAN:PVTime:RISE:X[:SCALe]:AUTO	Automatically sets the starting time (position) and scale (width) values.

Table 2-13: Display commands (cont.)

Command	Description
DISPlay:WLAN:PVTime:RISE:X[:SCALe]:OFFSet	Sets or queries the starting time value (offset).
DISPlay:WLAN:PVTime:RISE:X[:SCALe]:RESet	Resets the starting time (position) and scale (width) values.
DISPlay:WLAN:PVTime:WINDow:SELect:PLOT	Specifies or queries which view to use.
DISPlay:WLAN:PVTime:WINDow:TRACe:GRATicule:GRID:STATe	Shows or hides the graticule grid on the screen.
DISPlay:WLAN:PVTime:Y[:SCALe]	Sets or queries the vertical scale value.
DISPlay:WLAN:PVTime:Y[:SCALe]:AUTO	Automatically selects the vertical scale and position values.
DISPlay:WLAN:PVTime:Y[:SCALe]:OFFSet	Sets or queries the value of the vertical offset.
DISPlay:WLAN:PVTime:Y[:SCALe]:PDIVision	Sets or queries the value of the vertical scale.
DISPlay:WLAN:PVTime:Y[:SCALe]:RESet	Resets the vertical scale and position (offset) values.
DISPlay:WLAN:SUMMary:EVMUnits	Sets or queries the EVM Units setting.
DISPlay:LTE subgroup	
DISPlay:LTE:ACLR:MARKer:SHOW:STATe	Determines whether to show or hide the readout for the selected marker.
DISPlay:LTE:ACLR:RESet:SCALe	Resets the horizontal and vertical scale to default values.
DISPlay:LTE:ACLR:SHOW:LIMit	Sets or queries the appearance of the limits.
DISPlay:LTE:ACLR:WINDow:TRACe:GRATicule:GRID:STATe	Determines whether to show or hide the graticules.
DISPlay:LTE:ACLR:WINDow:TRACe:POWer:LEVEl:STATe	Determines whether to show or hide the power readout.
DISPlay:LTE:ACLR:X:SCALe	Sets or queries the horizontal range.
DISPLay:LTE:ACLR:X:SCALe:AUTO	Rescales the horizontal axis automatically to fit the waveform to the screen.
DISPlay:LTE:ACLR:X:SCALe:OFFSet	Sets or queries the center frequency of the trace.
DISPlay:LTE:ACLR:Y:SCALe:AUTO	Rescales the vertical scale automatically to fit the waveform on the screen.
DISPlay:LTE:ACLR:Y:SCALe:OFFSet	Sets or queries the vertical offset of the trace.
DISPlay:LTE:ACLR:Y:SCALe	Sets or queries the vertical range of the graph.
DISPlay:LTE:CHSPectrum:MARKer:SHOW:STATe	Determines whether to show or hide the readout for the selected marker.
DISPlay:LTE:CHSPectrum:RESet:SCALe	Resets the horizontal and vertical scale to default values.
DISPlay:LTE:CHSPectrum:WINDow:TRACe:GRATicule:GRID:STATe	Determines whether to show or hide the graticule.
DISPlay:LTE:CHSPectrum:X:SCALe	Sets or queries the horizontal range.
DISPLay:LTE:CHSpectrum:X:SCALe:AUTO	Rescales the horizontal axis automatically to fit the waveform to the screen.
DISPlay:LTE:CHSPectrum:X:SCALe:OFFSet	Sets or queries the center frequency of the trace.
DISPlay:LTE:CHSPectrum:Y:SCALe	Sets or queries the vertical range of the graph.
DISPlay:LTE:CHSPectrum:Y:SCALe:AUTO	Rescales the vertical scale automatically to fit the waveform on the screen.
DISPlay:LTE:CHSPectrum:Y:SCALe:OFFSet	Sets or queries the vertical offset of the trace.
DISPlay:LTE:CONSte:SCALe	Sets or queries the scale (zoom).
DISPlay:LTE:CONSte:X:OFFSet	Sets or queries the horizontal offset.
DISPlay:LTE:CONSte:Y:OFFSet	Sets or queries the vertical offset.

Table 2-13: Display commands (cont.)

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

Initiate Commands

Use the INITiate commands to control the acquisition of data.

Table 2-14: Initiate commands

Command	Description
INITiate:CONTinuous	Sets or queries whether to acquire data continuously.
INITiate[:IMMediate]	Starts input signal acquisition.
INITiate:RESume	Starts data acquisition.

Input Commands

Use the INPut commands to control the characteristics of the signal input. These commands are only available when an RSA306, RSA306B, RSA500A series, RSA600A series, MDO4000B series, or an MDO4000C series instrument is connected.

Table 2-15: Input commands

Command	Description
INPut:CORRection:EXTernal:EDIT<x>:INTerpolation	Sets or queries the interpolation setting to use with the indicated external gain table.
INPut:CORRection:EXTernal:EDIT<x>:NEW	Creates the indicated external loss table.
INPut:CORRection:EXTernal:TYPE	Sets or queries the data type to use when applying the external loss table corrections.
INPut:{MLEVel|RLEVel}	Sets or queries the reference level.

Mass Memory Commands

Use the MMEMory commands to manipulate files on the mass memory devices.

For the trace specifier TRACe<x>, refer to *Trace Mnemonics.* (See page 2-99.)

If the file extension is not included, it is assumed to be .tiq.

Table 2-16: Mass memory (MMEMory) commands

Command	Description
MMEMory:{AM|FM|PM} subgroup	**AM/FM/PM measurement**
MMEMory:{AM|FM|PM}:LOAD:TRACe	Loads trace data from the specified file.
MMEMory:Noise subgroup	**Noise Figure and Gain measurements**
MMEMory:NOISe:GAIN:LOAD:TRACe	Loads the Noise Gain trace data from the specified file.
MMEMory:NOISe:GAIN:SHOW:TRACe	Show or hide the Noise Gain trace from the specified file.
MMEMory:NOISe:GAIN:STORe:TRACe	Stores the Noise Gain trace data in the specified file.
MMEMory:Noise:LOAD:ENRCALibration	Loads the specified Noise Figure ENR table.
MMEMory:Noise:LOAD:FREQTABle	Loads the specified Noise Figure Frequency table.
MMEMory:Noise:STORe:ENRCALibration	Saves the Noise Figure ENR table to the specified location.
MMEMory:Noise:STORe:FREQTABle	Saves the Noise Figure Frequency table to the specified location.
MMEMory:{AM|FM|PM}:SHOW:TRACe<x>	Enables display of a recalled trace file.
MMEMory:{AM|FM|PM}:STORe:TRACe	Stores trace data in the specified file.
MMEMory:AUDio subgroup	**Audio measurements**
MMEMory:AUDio:LOAD:FILTer	Loads a user-defined audio filter from the specified file.
MMEMory:AVTime subgroup	**Amplitude versus Time measurement**
MMEMory:AVTime:LOAD:TRACe<x>	Loads trace data from the specified file.
MMEMory:AVTime:SHOW:TRACe<x>	Enables display of a recalled trace file.
MMEMory:AVTime:STORe:TRACe<x>	Stores trace data in the specified file.
MMEMory:CALibration subgroup (Requires an RSA306, RSA306B, RSA500A series, RSA600A series, MDO4000B series, or MDO4000C series instrument be connected.)	**Measurement calibration**
MMEMory:CALibration:LOAD:CORRection:EXTernal:EDIT<x>	Loads the external loss table data from a specified file.
MMEMory:CALibration:STORe:CORRection:EXTernal:EDIT<x>	Stores an external loss table to a specified file.
MMEMory:DDEMod subgroup	**DPX spectrum measurement**
MMEMory:DDEMod:LOAD:FILTer:MEASurement:USER<x>	Loads a user-defined measurement filter from the specified file.
MMEMory:DDEMod:LOAD:FILTer:MEASurement:UOTHer	Loads a user-defined measurement filter from the specified file.

Table 2-16: Mass memory (MMEMory) commands (cont.)

Command	Description
MMEMory:DDEMod:LOAD:FILTer:REFerence:USER<x>	Loads the specified user-defined reference filter.
MMEMory:DDEMod:LOAD:FILTer:REFerence:UOTHer	Loads the specified user-defined reference filter
MMEMory:DDEMod:LOAD:SYMBol:MAP	Loads the specified symbol map filename.
MMEMory:DPX subgroup	**DPX spectrum measurement**
MMEMory:DPX:TDM:FREQuency:TXBer:LOAD:PATTern	Loads the Tx BER User data pattern from the specified file.
MMEMory:DPX:LOAD:TRACe<x>	Loads trace data from the specified file.
MMEMory:DPX:SHOW:TRACe<x>	Enables display of a recalled trace file.
MMEMory:DPX:STORe:TRACe<x>	Stores trace data in the specified file.
MMEMory:{FSETtling|PSETtling} subgroup	**Frequency and Phase Settling time measurements**
[MMEMory:{FSETtling|PSETtling}:LOAD:TRACe<x>	Loads the Frequency or Phase Settling Time trace data from the specified file.
[MMEMory:{FSETtling|PSETtling}:SHOW:TRACe<x>	Enables display of a recalled Frequency or Phase Settling trace file in Trace<x>.
[MMEMory:{FSETtling|PSETtling}:STORe:TRACe<x>	Stores the Frequency or Phase Settling trace data in the specified file.
MMEMory:FVTime subgroup	**Frequency versus Time measurement**
MMEMory:FVTime:LOAD:TRACe	Loads trace data from the specified file.
MMEMory:FVTIME:SHOW:TRACe<x>	Enables display of a recalled trace file.
MMEMory:FVTime:STORe:TRACe	Stores trace data in the specified file.
MMEMory:IQVTime subgroup	**RF I&Q versus Time measurement**
MMEMory:IQVTime:LOAD:TRACe:I	Loads I trace data from the specified file.
MMEMory:IQVTime:LOAD:TRACe:Q	Loads Q trace data from the specified file.
MMEMory:IQVTIME:SHOW:TRACe:I	Loads trace data from the specified file.
MMEMory:IQVTIME:SHOW:TRACe<x>:Q	Enables display of a recalled trace file.
MMEMory:IQVTime:STORe:TRACe:I	Stores I trace data in the specified file.
MMEMory:IQVTime:STORe:TRACe:Q	Stores Q trace data in the specified file.
MMEMory:LTE subgroup	
MMEMory:LTE:ACLR:LOAD:TABLe	Loads the LTE ACLR table containing the limits for enabled ranges from the specified .cvs file.
MMEMory:LTE:ACLR:STORe:TABLe	Stores the LTE ACLR table containing the limits for enabled ranges in a specified .cvs file, allowing you to export the file into Microsoft Excel or other database system.
MMEMory:LOAD subgroup	**Load commands**
MMEMory:LOAD:IQ	Loads time-domain IQ waveform from a specified file.
MMEMory:LOAD:STATe	Loads the instrument setup from a specified file.
MMEMory:LOAD:TRACe	Loads the trace data from a specified file.

Table 2-16: Mass memory (MMEMory) commands (cont.)

Command	Description
MMEMory:OFDM CONSTe subgroup	**OFDM Constellation commands**
MMEMory:OFDM:CONSte:LOAD:TRACe	Loads trace data from the specified file.
MMEMory:OFDM:CONSTe:SHOW:TRACe	Enables the display of a recalled trace file.
MMEMory:OFDM:CONSte:STORe:TRACe	Stores trace data in the specified file.
MMEMory:PHVTime subgroup	**Phase versus Time measurement**
MMEMory:PHVTime:LOAD:TRACe	Loads trace data from the specified file.
MMEMory:PHVTime:SHOW:TRACe	Enables display of a recalled trace file.
MMEMory:PHVTime:STORe:TRACe	Stores trace data in the specified file.
MMEMory:PNOise subgroup	**Phase noise measurement**
MMEMory:PNOise:LOAD:TRACe<x>	Loads the Phase Noise trace data from the specified file.
MMEMory:PNOise:SHOW:TRACe<x>	Enables display of a recalled Phase Noise trace file.
MMEMory:PNOise:STORe:TRACe<x>	Stores the Phase Noise trace data in the specified file.
MMEMory:SEM subgroup	**Spectral Emissions Mask measurement**
MMEMory:SEM:LOAD:TABLe	Loads the SEM table from the specified file.
MMEMory:SEM:STORe:TABLe	Stores the SEM table in the specified file.
MMEMory:SGRam subgroup	**Spurious measurement**
MMEMory:SGRam:LOAD:TRACe	Loads the trace data from a specified file.
MMEMory:SGRam:SHOW:TRACe	Enables display of a recalled trace file.
MMEMory:SGRam:STORe:TRACe	Stores the Spectrogram trace data in the specified file.
MMEMory:SPECtrum subgroup	**Spectrum measurement**
MMEMory:SPECtrum:LOAD:TRACe	Loads the trace data from a specified file.
MMEMory:SPECtrum:SHOW:TRACe<x>	Enables display of a recalled trace file.
MMEMory:SPECtrum:STORe:TRACe<x>	Stores the spectrum trace data in the specified file.
MMEMory:SPURious subgroup	**Spurious measurement**
MMEMory:SPURious:LOAD:TABLe	Loads the spurious table from the specified file.
MMEMory:SPURious:STORe:TABLe	Stores the spurious table in the specified file.
MMEMory:STORe subgroup	**Store commands**
MMEMory:STORe:IQ	Saves time-domain IQ waveform in the acquisition memory to a specified file.
MMEMory:STORe:IQ:CSV	Saves time-domain IQ waveform in the acquisition memory to a specified file in the CSV (Comma Separated Values) format.
MMEMory:STORe:IQ:MAT	Saves time-domain IQ waveform in the acquisition memory to a specified file in the MATLAB format.
MMEMory:STORe:IQ:SELEct:DATA	Sets or queries the type of data stored in an acquisition data file.
MMEMory:STORe:IQ:SELEct:FRAMes	Sets or queries which frames to save when saving TIQ acquisition data files.
MMEMory:STORe:IQ:SELEct:LENGth	Sets or queries what portion of each record is saved when saving acquisition data.
MMEMory:STORe:MSTate	Stores the measurement parameters in a specified file.

Table 2-16: Mass memory (MMEMory) commands (cont.)

Command	Description
MMEMory:STORe:RESults	Stores the measurement results including measurement parameters and trace data to a specified file.
MMEMory:STORe:SCReen	Stores the measurement results in a specified file.
MMEMory:STORe:STATe	Stores the instrument setup in a specified file.
MMEMory:STORe:TRACe	Stores trace data in a specified file.
MMEMory:TOVerview subgroup	**Spurious measurement**
MMEMory:TOVerview:LOAD:TRACe1	Loads the trace data from a specified file.
MMEMory:TOVerview:SHOW:TRACe1	Enables display of a recalled trace file.
MMEMory:TOVerview:STORe:TRACe1	Stores the selected trace data into the specified file.
MMEMory:WLAN subbroup	**Wireless LAN measurements**
MMEMory:WLAN:CRESponse:LOAD:TRACe<x>	Loads the selected WLAN Channel Response trace data from the specified file.
MMEMory:WLAN:CRESponse:SHOW:TRACe<x>	Enables the display of a recalled WLAN Channel Response trace.
MMEMory:WLAN:CRESponse:STORe:TRACe<x>	Stores the WLAN Channel Response measurement trace data in the specified file.

Specifying the File

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

- You can omit the file extension to load and store data. The measurement-specific extension is automatically added.
- You can use the absolute path to specify the file name. For example, specify the *SAMPLE1* file in the *My Documents* folder on the C drive as "`C:\My Documents\SAMPLE1`".
- If you omit the directory path, the default path is used, which is *C:\Program Files\Tektronix\SignalVu-PC* initially.

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

Sense Commands

Use the SENSe commands to set up detailed measurement conditions.

If the file extension is not included, it is assumed to be .tiq.

Table 2-17: Sense commands

Command	Description
[SENSe] basic command subgroup	**General analysis parameter control**
[SENSe]:ACQuisition:{BANDwidth|BWIDth}	Sets or queries the acquisition bandwidth.
[SENSe] Bluetooth command subgroup	**Bluetooth measurements**
[SENSe]:BLUEtooth:ANALysis:LENGth	Sets or returns the analysis length.
[SENSe]:BLUEtooth:ANALysis:LENGth:ACTual?	Sets the actual analysis length.
[SENSe]:BLUEtooth:ANALysis:LENGth:AUTO	Sets or queries the length mode.
[SENSe]:BLUEtooth:ANALysis:MEAS[:BANDwidth|BWIDth]	Sets or returns the measurement bandwidth.
[SENSe]:BLUEtooth:ANALysis:MEAS:SOURce	Sets or returns the analysis measurement source.
[SENSe]:BLUEtooth:ANALysis:OFFSet	Sets or returns the symbol analysis offset.
[SENSe]:BLUEtooth:ANALysis:OFFSet:AUTO	Determines whether to set the carrier frequency error automatically or manually.
[SENSe]:BLUEtooth:ANALysis:TIMe:UNITs	Sets the time units.
[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):MODE	Sets or returns the measurement bandwidth mode.
[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):TINTerval	Sets or returns the measurement bandwidth frequency span.
[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):TINTerval:AUTO	Sets or returns the bandwidth to Auto or Manual.
[SENSe]:BLUEtooth:CONStellation:PREFs:GRATicule:SHOW	Show or hides the graticule state.
[SENSe]:BLUEtooth:CONStellation:PREFs:MARKERS:SHOW	Show or hides the marker readout.
[SENSe]:BLUEtooth:CONStellation:PREFs:RADIx	Sets or returns the radix of the marker readout.
[SENSe]:BLUEtooth:CONStellation:TRACE<x>:CONTent	Sets or returns the appearance of the specified trace (Trace 1 or Trace 2).
[SENSe]:BLUEtooth:CONStellation:TRACe<x>:FREEze	Sets or freezes the updates of the specified trace (Trace 1 or Trace 2).
[SENSe]:BLUEtooth:CONSTellation:TRACE<x>:POINtsPerSymbol	Sets or returns the number of points per symbol (how many points to use between symbols) for the specified trace (Trace 1 or Trace 2).
[SENSe]:BLUEtooth:CONStellation:TRACe<x>:SHOW	Shows or hides the specified trace (Trace 1 or Trace 2).
[SENSe]:BLUEtooth:DELTa:AVERage:FONE	Sets or returns the average frequency for the low deviation pattern.
[SENSe]:BLUEtooth:DELTa:AVERage:FTWO	Sets or returns the average frequency for the high deviation pattern.
[SENSe]:BLUEtooth:FDVTime:TRACE:OCTEt?	Returns the selected octet number on the Frequency Deviation vs. Time display.

Table 2-17: Sense commands (cont.)

Command	Description
[SENSe]:BLUEtooth:FDVTime:TRACE:OCTEt:TOTAL:COUNt?	Returns the total octet value to be selected on the Frequency Deviation vs. Time display.
[SENSe]:BLUEtooth:FDVTime:TRACE:VIEW:MODE	Sets or queries view mode in the Frequency Deviation vs. Time display.
[SENSe]:BLUEtooth:FILTer:ALPHa	Sets or returns the alpha filter factor (α/BT).
[SENSe]:BLUEtooth:FILTer:MEASurement	Sets or returns the measurement filter setting.
[SENSe]:BLUEtooth:FILTer:REFerence	Sets or returns the reference filter setting.

Table 2-17: Sense commands (cont.)

Command	Description
[SENSe]:BLUEtooth:FREQuency:DEViation	ets or returns the frequency deviation.
[SENSe]:BLUEtooth:FREQuency:DEViation:AUTO	Determines whether to detect the frequency deviation automatically or manually.
[SENSe]:BLUEtooth:FREQuency:ERRor	Sets or returns the frequency error in the Bluetooth analysis parameters (when Auto is off).
[SENSe]:BLUEtooth:FREQuency:ERRor:AUTO	Determines whether the frequency error is set to Auto or Manual.
[SENSe]:BLUEtooth:INBEmissions:POWEr:LIMIt<x>	Sets or queries the limits of the Bluetooth InBand Emission settings.
[SENSe]:BLUEtooth:POWer:CLASs	Sets or queries the power class of the Bluetooth standard.
[SENSe]:BLUEtooth:STANdard	Sets or queries the Bluetooth standard.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FONE:AVERage:MAX	Sets or returns the maximum limit of the Basic Rate ΔF1 average.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FONE:AVERage:MIN	Sets or returns the minimum limit of the Basic Rate ΔF1 average.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FTWO:PERcent	Sets or returns the limit of the Basic Rate ΔF2 percentage.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FTWO:RESult	Sets or returns the limit of the Basic Rate ΔF2 result.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:DRIFt:FNFZero	Sets or returns the limit of the Basic Rate maximum drift fn–f0.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:FREQuency:OFFSet	Sets or queries the limit of the Basic Rate maximum frequency offset.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:FREQuency:OFFSet:PREAmBDR	Sets or queries the limit of the Basic Rate maximum frequency offset value from the preamble region.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:AVERage:MAX	Sets or queries the limit of the Basic Rate Power Class 1 maximum average output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:AVERage:MIN	Sets or queries the limit of the Basic Rate Power Class 1 minimum average output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:PEAK:MAX	Sets or queries the limit of the Basic Rate Power Class 1 maximum peak output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:PEAK:MIN	Sets or queries the limit of the Basic Rate Power Class 1 minimum peak output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:AVERage:MAX	Sets or queries the limit of the Basic Rate Power Class 3 maximum average output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:AVERage:MIN	Sets or queries the limit of the Basic Rate Power Class 3 minimum average output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:PEAK:MAX	Sets or queries the limit of the Basic Rate Power Class 3 maximum peak output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:PEAK:MIN	Sets or queries the limit of the Basic Rate Power Class 3 minimum peak output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:AVERage:MAX	Sets or queries the limit of the Basic Rate Power Class 2 maximum average output power value.

Table 2-17: Sense commands (cont.)

Command	Description
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:AVERage:MIN	Sets or queries the limit of the Basic Rate Power Class 2 minimum average output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:PEAK:MAX	Sets or queries the limit of the Basic Rate Power Class 2 maximum peak output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:PEAK:MIN	Sets or queries the limit of the Basic Rate Power Class 2 minimum peak output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FONE:AVERage:MAX	Sets or returns the maximum limit of the Low Energy ΔF1 average.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FONE:AVERage:MIN	Sets or returns the minimum limit of the Low Energy ΔF1 average.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FTWO:PERcent	Sets or returns the limit of the Low Energy ΔF2 percentage.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FTWO:RESult	Sets or returns the limit of the Low Energy ΔF2 result.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:DRIFt:FNFZero	Sets or returns the limit of the Low Energy maximum drift fn–f0.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:FREQuency:OFFSet	Sets or returns the limit of the Low Energy maximum frequency offset.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:FREQuency:OFFSet:PREAmble	Sets or returns the limit of the Low Energy maximum frequency offset value from the preamble region.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:AVERage:MAX	Sets or returns the limit of the Low Energy maximum average output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:AVERage:MIN	Sets or returns the limit of the Low Energy minimum average output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:PEAK:MAX	Sets or returns the limit of the Low Energy maximum peak output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:PEAK:MIN	Sets or returns the limit of the Low Energy minimum peak output power value.
[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:F1FZero	Sets or returns the limit of the maximum fn–fn0 drift in the packet.
[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:FNFN5	Sets or returns the limit of the maximum fn–fn5 drift in the packet.
[SENSe]:BLUEtooth:SYMBol:POINts	Sets or returns the number of points per symbol (how many points to use between symbols).
[SENSe]:BLUEtooth:TEST:PATTern	Sets or returns the test pattern type in the Bluetooth displays.
[SENSe]:BLUEtooth:TEST:PATTern:AUTO:DETEct	Sets or returns the auto detect setting for the Bluetooth displays.
[SENSe]:BLUEtooth:TIME:ZERO:REFERENCE	Sets or returns the Time Zero Reference setting in the Bluetooth displays.
[SENSe]:BLUEtooth:TRACe:TYPE	Sets or returns the trace type.
[SENSe]: NOISe subgroup	**Noise Figure and Gain measurements**
[SENSe]:NOISe:AVERage:COUNt	Sets or queries the acquisition counts over which the noise average is computed.

Table 2-17: Sense commands (cont.)

Command	Description
[SENSe]:NOISe:BANDwidth	Sets or queries the resolution bandwidth (RBW).
[SENSE]:NOISe:CONTrol:SETTling:OFF[:TIME]	Sets or queries the analyzer off settle time.
[SENSE]:NOISe:CONTrol:SETTling:ON[:TIME]	Sets or queries the analyzer on settle time.
[SENSe]:NOISe:CORRection:ENR:CONSTant	Sets and queries the excess noise ratio (ENR) constant.
[SENSe]:NOISe:CORRection:ENR:MODE	Sets and queries the Noise Source mode for the noise displays.
[SENSe]:NOISe:CORRection:ENR:TABLe:DATA?	Queries the Noise Source model, serial number, and table values for frequency and ENR for the noise displays.
[SENSe]:NOISe:CORRection:ENR:TABLe:ID	Sets or queries the Noise Source Model for the noise displays.
[SENSe]:NOISe:CORRection:ENR:TABLe:NEW	Sets the Noise Source Model for the noise displays.
[SENSe]:NOISe:CORRection:ENR:TABLe:SERial	Sets or queries the Noise Source serial number for the noise displays.
[SENSe]:NOISe:CORRection:TCOLd:ADUT:UREFerence	Sets or queries the analyzer to use the reference temperature (K) for the loss at the DUT output in the noise displays.
[SENSe]:NOISe:CORRection:TCOLd:ADUT[:VALue]	Sets or queries the temperature value (K) for the loss at the DUT output in the noise displays.
[SENSe]:NOISe:CORRection:TCOLd:BDUT:UREFerence	Sets or queries the analyzer to use the reference temperature (K) for the loss at the DUT input in the noise displays.
[SENSe]:NOISe:CORRection:TCOLd:BDUT[:VALue]	Sets or queries the temperature value (K) for the loss at the DUT input in the noise displays.
[SENSe]:NOISe:CORRection:TCOLd[:REFerence]	Sets or queries the reference temperature value (K) in the display.
[SENSe]:NOISe:FIGure:POINTs:COUNt	Sets the number of sample points for the display.
[SENSe]:NOISe:FREQuency:CENTer	Sets or queries the center frequency value (Hz).
[SENSe]:NOISe:FREQuency:MODE	Sets or queries the frequency mode in the display.
[SENSe]:NOISe:FREQuency:SPAN	Sets or queries the frequency Span value (Hz).
[SENSe]:NOISe:FREQuency:STARt	Sets or queries the Start Frequency value.
[SENSe]:NOISe:FREQuency:STEPs	Sets or queries the frequency Steps value.
[SENSe]:NOISe:FREQuency:STOP	Sets or queries the Stop Frequency value.
[SENSe]:NOISe:FREQuency:TABLe:DATA?	Queries the data in the Frequency Table.
[SENSe]:NOISe:FREQuency:TABLe:NEW	Creates a new Frequency Table with the specified values for the noise display.
[SENSe]:OBWidth:BOBW:XDBM:MEASurement:DIREction	Sets or queries the measurement direction.
[SENSe]:OBWidth:BOBW:XDBM:RANGe	Sets or queries the measurement range.
[SENSe]:P25 subgroup	**APCO P25 measurements**
[SENSe]:P25:ANALysis:OFFSet	Sets or queries the offset value.
[SENSe]:P25:ANALysis:OFFSet:AUTO	Sets or queries if the offset is set to Auto or if Auto is off.
[SENSe]:P25([:BANDwidth]|:BWIDth):MODE	Sets or queries the Measurement BW (bandwidth) (frequency span) mode as Auto, Manual, or Link to Span.
[SENSe]:P25([:BANDwidth]|:BWIDth):TINTerval	Sets or queries the Measurement BW (bandwidth) (frequency span).

Table 2-17: Sense commands (cont.)

Command	Description
[SENSe]:P25([:BANDwidth]|:BWIDth):TINTerval:AUTO	Sets or queries the Measurement BW (bandwidth) (frequency span) mode as Auto or Manual.
[SENSe]:P25:FILTer:MEASurement	Sets or queries the Measurement Filter setting.
[SENSe]:P25:FILTer:REFerence	Sets or queries the Reference Filter setting.
[SENSe]:P25:FREQuency:ERRor	Sets or queries the Frequency Offset (when Auto is off).
[SENSe]:P25:FREQuency:ERRor:AUTO	Sets or queries if the Frequency Error is set to Auto or if Auto is off.
[SENSe]:P25:MODUlation:CHANnel:([BANDwidth]|:BWIDth)]?	Queries the actual P25 Preset channel bandwidth.
[SENSe]:P25:MODUlation:STANdard	Sets or queries the modulation standard.
SENSe:P25:MODUlation:TYPE	Sets or queries the modulation type.
[SENSe]:P25:SRATe	Sets or queries the symbol rate value for measurements.
[SENSe]:P25:SUMMary:CLEar:RESults	Clears the results from the P25 Summary display.
[SENSe]:P25:SUMMary:LIMIt:FA	Sets or queries the On/Off state of limit comparison and the upper limit value for Operating Frequency Accuracy.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:AVERage:TERRor:ONE	Sets or queries the On/Off state of limit comparison and the upper limit value for Average t_error_1.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:AVERage:TERRor:ZERO	Sets or queries the On/Off state of limit comparison and the upper limit value for Average t_error_0.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:LOGIc:CHANnel:OFF:SLOT	Sets or queries the On/Off state of limit comparison and the upper limit value for HCPM Transmitter Logic channel offslot.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:LOGIc:CHANnel:PEAK:ACPR	Sets or queries the On/Off state of limit comparison and the upper limit value for HCPM Transmitter Logic channel Peak ACPR.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:NEGAtive:LO	Sets or queries the On/Off state of limit comparison and the lower limit value for Frequency Deviation, negative peak.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:NEGAtive:UP	Sets or queries the On/Off state of limit comparison and the upper limit value for Frequency Deviation, negative peak.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:POSItive:LO	Sets or queries the On/Off state of limit comparison and the lower limit value for Frequency Deviation, positive peak.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:POSItive:UP	Sets or queries for the On/Off state of limit comparison and the upper limit value for Frequency Deviation, positive peak.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PHASe[1/2]:F[1/2/3/4]:LO	Sets or queries the On/Off state of limit comparison and the lower limit value for Frequency Deviation, f1/f2/f3/f4 for Phase1 or Phase2.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PHASe[1/2]:F[1/2/3/4]:UP	Sets or queries the On/Off state of limit comparison and the upper limit value for Frequency Deviation, f1/f2/f3/f4 for Phase1 or Phase2.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:OFF	Sets or queries the On/Off state of limit comparison and the upper limit value for Pmax-off.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:ON	Sets or queries the On/Off state of limit comparison and the upper limit value for Pmax-on.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:SS	Sets or queries the On/Off state of limit comparison and the upper limit value for Pss-max.

Table 2-17: Sense commands (cont.)

Command	Description
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMIN:SS	Sets or queries the On/Off state of limit comparison and the upper limit value for Pss-min.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter:AVERage:ATTK:ENCOder	Sets or queries the On/Off state of limit comparison and the upper limit value for Transmitter Average Attack encoder.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter:AVERage:ATTK:ENCOder:BI	Sets or queries the On/Off state of limit comparison and the upper limit value for Transmitter Average Attack encoder, busy/idle.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter:AVERage:ATTK:POWEr	Sets or queries the On/Off state of limit comparison and the upper limit value for Transmitter Average Attack power.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter:AVERage:ATTK:POWEr:BI	Sets or queries for the On/Off state of limit comparison and the upper limit value for Transmitter Average Attack power, busy/idle.
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter:AVERage:THRUput:DLAY	Sets or queries the On/Off state of limit comparison and the upper limit value for Transmitter Average throughput delay.
[SENSe]:P25:SUMMary:LIMIt:MF	Sets or queries the On/Off state of limit comparison and the upper limit value for Modulation Fidelity.
[SENSe]:P25:SUMMary:LIMIt:RF:OUTPower	Sets or queries the On/Off state of limit comparison and the upper limit value for RF Output Power.
[SENSe]:P25:SUMMary:LIMIt:SRA	Sets or queries the On/Off state of limit comparison and the upper limit value for Symbol Rate Accuracy.
[SENSe]:P25:SYMBols:POINts	Sets or queries the Points/Symbols to display.
SENSe:P25:TIMe:UNITs	Sets or queries the time units as Seconds or Symbols in the display.
[SENSe]:P25:TIMe:UNITs	Sets or queries the fundamental unit of time for the P25 measurements.
[sense]:acquisition subgroup	**Acquisition measurements**
[SENSe]:ACQuisition:MODE	Sets or queries the acquisition mode.
[SENSe]:ACQuisition:OPTimization	Sets or queries the method of the global gain and input bandwidth optimization.
[SENSe]:ACQuisition:SAMPles	Sets or queries the acquisition samples.
[SENSe]:ACQuisition:SEConds	Sets or queries the acquisition length.
[SENSe]:ANALysis:ADVanced:DITHer	Determines whether to enable or disable dithering.
[SENSe]:ANALysis:ADVanced:DITHer:HWARe:STATus?	Returns the dithering hardware status.
[SENSe]:ANALysis:LENGth	Sets or queries the analysis length.
[SENSe]:ANALysis:LENGth:ACTual?	Queries the actual analysis length.
[SENSe]:ANALysis:LENGth:AUTO	Sets or queries whether to set the analysis length automatically.
[SENSe]:ANALysis:REFerence	Sets or queries the analysis time reference.
[SENSe]:ANALysis:STARt	Sets or queries the analysis offset time.
[SENSe]:ANALysis:STARt:AUTO	Sets or queries whether to set the analysis offset automatically.
[SENSe]:ANSPectrum:STARt:AUTO:METHod	Sets or queries the method used for computing auto analysis and spectrum offsets.
[SENSe]:MEASurement:FREQuency	Sets or queries the measurement frequency.
[SENSe]:MEASurement:FREQuency:CENTer:LOCK	Sets or queries locking the measurement center frequency.

Table 2-17: Sense commands (cont.)

Command	Description
[SENSe]:POWer:UNITs	Sets or queries the unit of power.
[SENSe]:SIGNalpath:PRIority	Sets or queries the signal path priority for the measurements (for RSA5000 series only).
[SENSe]:USETtings	Updates the analyzer settings.
[SENSe]:ACPower subgroup	**Channel power and ACPR measurement**
[SENSe]:ACPower:AVERage	Sets or queries how to average waveform.
[SENSe]:ACPower:AVERage:COUNt	Sets or queries the number of traces for averaging.
[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution]	Sets or queries the resolution bandwidth (RBW).
[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution]:ACTual?	Queries the actual resolution bandwidth (RBW).
[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution]:AUTO	Sets or queries whether to set the RBW automatically.
[SENSe]:ACPower:{BANDwidth|BWIDth}:VIDeo	Sets or queries the video bandwidth (VBW).
[SENSe]:ACPower:{BANDwidth|BWIDth}:VIDeo:STATe	Sets or queries whether to enable the video bandwidth (VBW).
[SENSe]:ACPower:CHANnel:{BANDwidth|BWIDth}	Sets or queries the channel bandwidth.
[SENSe]:ACPower:CHANnel:FILTer	Sets or queries the adjacent channel filter.
[SENSe]:ACPower:CHANnel:PAIRs	Sets or queries the number of adjacent channel pairs.
[SENSe]:ACPower:CHANnel:SPACing	Sets or queries the channel-to-channel spacing.
[SENSe]:ACPower:CHIPrate	Sets or queries the chip rate.
[SENSe]:ACPower:CLEar:RESults	Restarts the average trace.
[SENSe]:ACPower:FREQuency	Sets or queries the center frequency.
[SENSe]:ACPower:FREQuency:STEP	Sets or queries the frequency step size.
[SENSe]:ACPower:FREQuency:STEP:AUTO	Sets or queries whether to set the frequency step size automatically.
[SENSe]:ACPower:NFLoor:STATe	Sets or queries whether to enable the correction for noise floor.
[SENSe]:ACPower:POINts:COUNt	Sets or queries the number of trace points.
[SENSe]:ACPower:RRCRolloff	Sets or queries the filter parameter for the Root Raised Cosine filter.
[SENSe]:ACPower:SYMBrate	Sets or queries the symbol rate.
[SENSe]:{AM|FM|PM} subgroup (Option 21 only)	**AM/FM/PM measurement**
[SENSe]:{AM|FM|PM}:{BANDwidth|BWIDth}:MEASurement	Sets measurement bandwidth for the AM, FM, or PM demodulation to analyze.
[SENSe]:{AM|FM|PM}:CLEar:RESults	Clears measurement results.
[SENSe]:{AM|FM|PM}:{MTPoints|MAXTracepoints}	Sets or queries the maximum trace points.
[SENSe]:AM:DETect:AMPLitude	Sets or queries the carrier amplitude detection method.
[SENSe]:{FM|PM}:BURSt:THReshold	Sets or queries the threshold level to detect bursts.
[SENSe]:{FM|PM}:FREQuency:OFFSet	Sets or queries the carrier frequency offset in the FM/PM measurement.
[SENSe]:{FM|PM}:FREQuency:OFFSet:MARKer	Sets the frequency offset from selected marker or delta marker location.
[SENSe]:{FM|PM}:FREQuency:SEARch:AUTO	Sets or queries whether to detect the carrier frequency automatically.
[SENSe]:PM:PHASe:OFFSet	Sets the phase offset from the selected marker location.

Table 2-17: Sense commands (cont.)

Command	Description
[SENSe]:PM:PHASe:OFFSet:MARKer	Sets the carrier frequency offset from the selected marker.
[SENSe]:PM:PHASe:SEARch:AUTO	Sets or queries whether to detect the phase automatically or manually.
{SENSe]:AUDio subgroup	**Audio measurement**
[SENSe]:AUDio:AVERage:COUNt	Sets or queries the number of harmonics to average in the audio measurement.
[SENSe]:AUDio:AVERage:ENABle	Enables or disables averaging of harmonics in the audio measurement.
[SENSe]:AUDio:BANDwidth|BWIDth	Sets or queries the audio measurement bandwidth.
[SENSe]:AUDio:BANDwidth|BWIDth:RESolution	Sets or queries the resolution bandwidth for the audio measurement.
[SENSe]:AUDio:BANDwidth|BWIDth:RESolution:AUTO	Sets or queries the resolution bandwidth for the audio measurement.
[SENSe]:AUDio:BANDwidth|BWIDth:RESolution:SHAPe	Sets or queries the shape of the resolution bandwidth for the audio measurement.
[SENSe]:AUDio:CARRier:OFFSet	Sets or queries the carrier frequency offset.
[SENSe]:AUDio:CARRier:OFFSet:AUTO	Determines whether to set the carrier frequency error automatically or manually.
[SENSe]:AUDio:FILTer:DEEMphasis	Sets or queries the de-emphasis filter type.
[SENSe]:AUDio:FILTer:DEEMphasis:ENABle	Enables or disables the de-emphasis filter for the audio measurement.
[SENSe]:AUDio:FILTer:DEEMphasis:USER	Sets or queries a user-defined de-emphasis audio filter.
[SENSe]:AUDio:FILTer:HPF	Sets or queries the high-pass filter type.
[SENSe]:AUDio:FILTer:HPF:ENABle	Enables or disables the high-pass audio filter for the audio measurement.
[SENSe]:AUDio:FILTer:HPF:USER	Sets or queries a user-defined high-pass audio filter.
[SENSe]:AUDio:FILTer:DEEMphasis:USER	Sets or queries a user-defined high-pass audio filter.
[SENSe]:AUDio:FILTer:LPF	Sets or queries the low-pass filter type.
[SENSe]:AUDio:FILTer:LPF:ENABle	Enables or disables the low-pass filter for the audio measurement.
[SENSe]:AUDio:FILTer:LPF:USER	Sets or queries a user-defined low-pass audio filter.
[SENSe]:AUDio:FILTer:MODE:PREDefined	Selects a predefined audio filter or a filter stored on the instrument file.
[SENSe]:AUDio:FILTer:SOURce?	Queries the measurement filter from the user-defined source file.
[SENSe]:AUDio:FILTer:STANdard	Sets or queries the Standard audio filter.
[SENSe]:AUDio:FILTer:STANdard:ENABle	Enables or disables a Standard filter for the audio measurement.
[SENSe]:AUDio:FREQuency	Sets or queries the reference audio frequency.
[SENSe]:AUDio:FREQuency:AUTO	Determines whether to set the reference audio frequency automatically or manually.
[SENSe]:AUDio:HNOise:ENABle	Enables or disables the Hum and Noise information in the Audio Summary display.
[SENSe]:AUDio:HNOise:REFerence	Captures the current Hum and Noise information in the Audio Summary display.
[SENSe]:AUDio:SIGnal:TYPE	Sets or queries the audio signal type.
[SENSe]:AUDio:SPECtrum:CLEar:RESults	Clears the results of the audio spectrum measurement.

Table 2-17: Sense commands (cont.)

Command	Description
[SENSe]:AUDio:SPECtrum:POINts:COUNt	Sets or queries the number of trace points acquired for the audio spectrum display.
[SENSe]:AUDio:UNITs	Sets or queries the units of the audio spectrum measurement.
[SENSe]:AVTime subgroup	**Amplitude versus Time measurement**
[SENSe]:AVTime:{BANDwidth|BWIDth}	Sets or queries the time-domain bandwidth filter.
[SENSe]:AVTime:{BANDwidth|BWIDth}:ACTual?	Queries the actual resolution bandwidth (RBW).
[SENSe]:AVTime:CLEar:RESults	Restarts multi-trace functions (Average and Max/Min Hold).
[SENSe]:AVTime:MAXTracepoints	Sets or queries the maximum trace points.
[SENSe]:AVTime:METHod	Sets or queries the method to set the measurement bandwidth.
[SENSe]:AVTime:SPAN	Sets or queries the frequency span.
[SENSe]:DDEMod subgroup (Option 21 only)	**General purpose digital modulation measurements**
[SENSe]:DDEMod:ANALysis:LENGth	Sets or queries the analysis length.
[SENSe]:DDEMod:ANALysis:LENGth:ACTual?	Queries the actual analysis length.
[SENSe]:DDEMod:ANALysis:LENGth:AUTO	Sets or queries whether to set the analysis length automatically.
[SENSe]:DDEMod:{BANDwidth|BWIDth}:MODE	Sets or queries the measurement bandwidth mode.
[SENSe]:DDEMod:{BANDwidth|BWIDth}:TINTerval	Sets or queries the measurement bandwidth (frequency span).
[SENSe]:DDEMod:{BANDwidth|BWIDth}:TINTerval:AUTO	Sets the measurement bandwidth (frequency span) automatically.
[SENSe]:DDEMod:BURSt:DETect	Sets or queries how to detect bursts.
[SENSe]:DDEMod:BURSt:THReshold	Sets or queries the threshold level to determine a burst.
[SENSe]:DDEMod:CARRier:OFFSet	Sets or queries the carrier frequency offset.
[SENSe]:DDEMod:CARRier:OFFSet:AUTO	Sets or queries whether to detect the carrier frequency automatically.
[SENSe]:DDEMod:EQUalizer:AVAilable?	Determines if the equalizer is available.
[SENSe]:DDEMod:EQUalizer:CONVergence	Sets or queries the Convergence value (also known as tap update rate).
[SENSe]:DDEMod:EQUalizer:ENABle	Sets or queries the Equalizer state.
[SENSe]:DDEMod:EQUalizer:LENGth	Sets or queries the equalizer filter length.
[SENSe]:DDEMod:EQUalizer:MODE	Sets or queries the equalizer mode. The equalizer can in learning mode (train) or enabled (hold).
[SENSe]:DDEMod:EQUalizer:RESet	Resets the equalizer settings to default values.
[SENSe]:DDEMod:EQUalizer:TAPS	Sets or queries the number of filter coefficients for the equalizer.
[SENSe]:DDEMod:EQUalizer:TSRAtio	Sets or queries the Equalizer Taps/Symbol parameter.
[SENSe]:DDEMod:FILTer:ALPHa	Sets or queries the filter factor (a/BT).
[SENSe]:DDEMod:FILTer:MEASurement	Sets or queries the measurement filter.
[SENSe]:DDEMod:FILTer:REFerence	Sets or queries the reference filter.
[SENSe]:DDEMod:FREQuency:DEViation	Sets or queries the frequency deviation to determine two states for FSK.
[SENSe]:DDEMod:FREQuency:DEViation:AUTO	Sets or queries whether to detect the frequency deviation automatically.
[SENSe]:DDEMod:MAGNitude:NORMalize	Sets or queries the method for the magnitude normalization.

Table 2-17: Sense commands (cont.)

Command	Description
[SENSe]:DDEMod:MINDex	Sets or queries the modulation index for a CPM signal.
[SENSe]:DDEMod:MINDex:AUTO	Sets or queries whether to detect the modulation index automatically.
[SENSe]:DDEMod:MODulation:TYPE	Sets or queries the modulation type.
[SENSe]:DDEMod:PRESet	Presets the modulation analysis to a communication standard.
[SENSe]:DDEMod:RING:RADius	Sets or queries the APSK radius parameter.
[SENSe]:DDEMod:RING:ROTation	Sets or queries the APSK rotation parameter.
[SENSe]:DDEMod:SRATe	Sets or queries the symbol rate.
[SENSe]:DDEMod:SWAP:IQ	Sets or queries whether or not to swap I and Q data.
[SENSe]:DDEMod:SYMBol:HSSHift	Sets or queries the Q data half-symbol shift for OQPSK and SOQPSK signals.
[SENSe]:DDEMod:SYMBol:MAP:SOURce?	Specifies or queries the user symbol map file.
[SENSe]:DDEMod:SYMBol:MAP[:STATe]	Sets or queries whether to use the user symbol map.
[SENSe]:DDEMod:SYMBol:PLOT:POSition	Sets or queries the symbol point location on an SOQPSK waveform.
[SENSe]:DDEMod:SYMBol:POINts	Sets or queries the number of points per symbol.
[SENSe]:DDEMod:SYMBol:RATE:SEARch	Determines whether to enable a symbol rate search.
[SENSe]:DDEMod:SYNCh:WORD	Sets or queries whether to enable the synchronization word.
[SENSe]:DDEMod:SYNCh:WORD:SYMBol	Sets or queries the synchronization word.
[SENSe]:DDEMod:TIMe:UNITs	Sets or queries the fundamental unit of time.
[SENSe]:DPX subgroup	**DPX spectrum measurement**
[SENSe]:DPX:AUDio:DEMod:GAIN	Sets or queries the gain for the audio demodulation.
[SENSe]:DPX:AUDio:DEMod:RXBWidth	Sets or queries the receiver bandwidth for the audio demodulation.
[SENSe]:DPX:AUDio:DEMod:RXFRequency?	Queries the receiver frequency for the audio demodulation.
[SENSe]:DPX:AUDio:DEMod:STATe	Sets or queries whether to enable or disable the audio demodulation.
[SENSe]:DPX:AUDio:DEMod:TUNE	Sets or queries how to determine the tuning frequency.
[SENSe]:DPX:AUDio:DEMod:TYPE	Sets or queries the modulation type for the audio demodulation.
[SENSe]:DPX:{BANDwidth|BWIDth}:ACTual?	Sets or queries the resolution bandwidth (RBW).
[SENSe]:DPX:{BANDwidth|BWIDth}:RESolution	Sets or queries the resolution bandwidth.
[SENSe]:DPX:{BANDwidth|BWIDth}[:RESolution]:AUTO	Determines whether to set the resolution bandwidth (RBW) automatically or manually in the DPX spectrum measurement.
[SENSe]:DPX:FREQuency:SPAN:{BANDwidth|BWIDth}[:RESolution]:RATio	Sets or queries whether to set the RBW automatically.
[SENSe]:DPX:CLEar:RESults	Restarts multi-trace functions (Average and Max/Min Hold).
[SENSe]:DPX:COLor	Sets or queries the color palette of three-dimensional graphs.
[SENSe]:DPX:COLor:MAXimum	Sets or queries the maximum value of the color axis.
[SENSe]:DPX:COLor:MINimum	Sets or queries the minimum value of the color axis.
[SENSe]:DPX:DDENsity:MEASurement:SHOW	Sets or queries the Show measurement value (measurement always visible for DPX Trigger).

Table 2-17: Sense commands (cont.)

Command	Description
[SENSe]:DPX:DGRam:COLor:MAXimum	Sets or queries the maximum value of the color axis in the DPXogram measurement.
[SENSe]:DPX:DGRam:COLor:MINimum	Sets or queries the minimum value of the color axis in the DPXogram measurement.
[SENSe]:DPX:DGRam:POINts:COUNt	Sets the number of trace points acquired for the DPXogram display.
[SENSe]:DPX:DGRam:TIME:CAPacity?	Returns the time capacity of the DPXogram measurement.
[SENSe]:DPX:DGRam:TIME:RESolution	Sets the time resolution of the DPXogram measurement.
[SENSe]:DPX:DGRam:TIME:RESolution:AUTO	Sets the time resolution automatically or manually in the DPXogram measurement.
[SENSe]:DPX:DGRam:WATerfall:DIRection	Sets or queries the direction of the Waterfall (DPXogram) display.
[SENSe]:DPX:DGRam:WATerfall:ENABle	Sets or queries the Waterfall (DPXogram) display.
[SENSe]:DPX:DGRam:Y[:SCALe]:AUTO	Rescales the height axis automatically to fit the waveform to the screen in the DPXogram display.
[SENSe]:DPX:DGRam:Y[:SCALe]:RESet	Resets the height scale of the DPXogram display.
[SENSe]:DPX:DWELl	Sets or queries the value of the Dwell time for the DPX spectrum measurement.
[SENSe]:DPX:DWELl:AUTO	Sets the value of the Dwell time automatically.
[SENSe]:DPX:PLOT	Sets or queries the plot in the DPX spectrum measurement.
[SENSe]:DPX:POINts:COUNt	Sets or queries the number of DPX trace points.
[SENSe]:DPX:FREQuency:CENTer	Sets or queries the center frequency.
[SENSe]:DPX:FREQuency:SPAN	Sets or queries the frequency span.
[SENSe]:DPX:FREQuency:STARt	Sets or queries the measurement start frequency.
[SENSe]:DPX:FREQuency:STEP	Sets or queries the frequency step size.
[SENSe]:DPX:FREQuency:STEP:AUTO	Sets or queries whether to set the frequency step size automatically.
[SENSe]:DPX:FREQuency:STOP	Sets or queries the measurement stop frequency.
[SENSe]:DPX:TDM:{BANDwidth|BWIDth}	Sets or queries the value of the measurement bandwidth in the DPX measurement.
[SENSe]:DPX:TDM:{BANDwidth|BWIDth}:ACTual?	Queries the actual bandwidth in the DPX measurement.
[SENSe]:DPX:TDM:FREQuency:CENTer	Sets or queries the center frequency in the DPX spectrum measurement.
[SENSe]:DPX:TDM:FREQuency:TXBer:CLEar:RESults	Clears the measurement results of the active Tx BER measurement.
[SENSe]:DPX:TDM:FREQuency:TXBer:ENABle	Sets or queries the Tx BER measurement.
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern	Sets or queries the Tx BER data pattern.
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:INVert	Sets or queries the data pattern Invert setting.
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:REPeat	Sets or queries the test duration for a User Pattern.
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:REVerse	Sets or queries whether the data pattern is in normal order or reversed.

Table 2-17: Sense commands (cont.)

Command	Description
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:SOURce?	Returns the path and file name of the user-defined pattern file.
[SENSe]:DPX:TDM:RBW	Sets or queries the value of the RBW filter.
[SENSe]:DPX:TDM:RBW:STATe	Determines whether to set the resolution bandwidth (RBW) automatically or manually.
[SENSe]:DPX:TDM:SWEep:TIME	Sets or queries the sweep time.
[SENSe]:DPX:TDM:TMOTion	Sets or queries the scroll direction.
[SENSe]:{FSETtling|PSETtling} subgroup	**Settling time measurements**
[SENSe]:{FSETtling|PSETtling}:{BANDwidth|BWIDTH}	Sets or queries the measurement bandwidth for the Settling Time measurements.
[SENSe]:{FSETtling|PSETtling}:{BANDwidth|BWIDTH}:ACTual?	Queries the actual measurement bandwidth in the Settling Time measurements.
[SENSe]:{FSETtling|PSETtling}:FREQuency:CENTer	Queries the actual measurement bandwidth in the Settling Time measurements.
[SENSe]:{FSETtling|PSETtling}:FREQuency:OFFSet	Sets or queries the offset frequency.
[SENSe]:{FSETtling|PSETtling}:LENGth	Sets or queries the measurement length.
[SENSe]:{FSETtling|PSETtling}:LENGth:ACTual?	Queries the actual measurement length for the Settling Time measurements.
[SENSe]:{FSETtling|PSETtling}:MASK:LIMit	Sets or queries the frequency or phase limit for the specified region.
[SENSe]:{FSETtling|PSETtling}:MASK:STATe	Sets or queries mask test state.
[SENSe]:{FSETtling|PSETtling}:MASK:TIME:REFerence	Sets or queries the mask time reference.
[SENSe]:{FSETtling|PSETtling}:MASK:TIME:STARt	Sets or queries the mask region start time in seconds.
[SENSe]:{FSETtling|PSETtling}:MASK:TIME:STOP	Sets or queries the mask region stop time in seconds.
[SENSe]:{FSETtling|PSETtling}:MASK:TIME:STOP	Sets or queries the mask region stop time in seconds.
[SENSe]:{FSETtling|PSETtling}:MAXTracepoints	Sets or queries the maximum trace points in the Settling Time measurements.
[SENSe]:{FSETtling|PSETtling}:SDURation:MINimum	Sets or queries the minimum settled duration in seconds.
[SENSe]:{FSETtling|PSETtling}:TARGet:REFerence	Sets or queries the target reference frequency.
[SENSe]:{FSETtling|PSETtling}:TOLerance	Sets or queries the frequency or phase range into which the signal must settle.
[SENSe]:FVTime subgroup	**Frequency versus Time measurement**
[SENSe]:FVTime:CLEar:RESults	Restarts multi-trace functions (Average and Max/Min Hold).
[SENSe]:FVTime:FREQuency:CENTer	Sets or queries the center frequency.
[SENSe]:FVTime:FREQuency:SPAN	Sets or queries the frequency span.
[SENSe]:FVTime:FREQuency:STARt	Sets or queries the measurement start frequency.
[SENSe]:FVTime:FREQuency:STEP	Sets or queries the frequency step size.
[SENSe]:FVTime:FREQuency:STEP:AUTO	Sets or queries whether to set the frequency step size automatically.
[SENSe]:FVTime:FREQuency:STOP	Sets or queries the measurement stop frequency.

Table 2-17: Sense commands (cont.)

Command	Description
[SENSe]:FVTime:MAXTracepoints	Sets or queries the maximum trace points.
[SENSe]:IQVTime subgroup	**RF I&Q versus Time measurement**
[SENSe]:IQVTime:CLEar:RESults	Restarts multi-trace functions (Average and Max/Min Hold).
[SENSe]:IQVTime:FREQuency:CENTer	Sets or queries the center frequency.
[SENSe]:IQVTime:FREQuency:SPAN	Sets or queries the frequency span.
[SENSe]:IQVTime:FREQuency:STARt	Sets or queries the measurement start frequency.
[SENSe]:IQVTime:FREQuency:STEP	Sets or queries the frequency step size.
[SENSe]:IQVTime:FREQuency:STEP:AUTO	Sets or queries whether to set the frequency step size automatically.
[SENSe]:IQVTime:FREQuency:STOP	Sets or queries the measurement stop frequency.
[SENSe]:IQVTime:MAXTracepoints	Sets or queries the maximum trace points.
[SENSe]:MCPower subgroup	**MCPR measurement**
[SENSe]:MCPower:AVERage	Sets or queries how to average waveform.
[SENSe]:MCPower:AVERage:COUNt	Sets or queries the number of waveforms for average.
[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution]	Sets or queries the resolution bandwidth (RBW).
[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution]:ACTual?	Queries the actual resolution bandwidth (RBW).
[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution]:AUTO	Sets or queries whether to set the RBW automatically.
[SENSe]:MCPower:{BANDwidth|BWIDth}:VIDeo	Sets or queries the video bandwidth (VBW).
[SENSe]:MCPower:{BANDwidth|BWIDth}:VIDeo:STATe	Sets or queries whether to enable the video bandwidth (VBW).
[SENSe]:MCPower:CHANnel:ADJacent:ADD	Adds a pair of upper and lower adjacent channels.
[SENSe]:MCPower:CHANnel:ADJacent:DELete	Deletes a selected adjacent channel.
[SENSe]:MCPower:CHANnel:ADJacent:LIMIts<x>:ENABle?	Queries whether the adjacent channel is selected or not in the P25 MCPR display.
[SENSe]:MCPower:CHANnel:ADJacent:LIMIts<x>	Sets or queries the selected adjacent upper and lower limit values in the P25 MCPR display.
[SENSe]:MCPower:CHANnel:FILTer	Sets or queries the measurement filter.
[SENSe]:MCPower:CHANnel:MAIN:{BANDwidth|BWIDth}	Sets or queries the frequency bandwidth of the main channels.
[SENSe]:MCPower:CHANnel:MAIN:COUNt	Sets or queries the number of main channels.
[SENSe]:MCPower:CHANnel:MAIN:INACtive	Makes a main channel inactive or queries the inactive main channels.
[SENSe]:MCPower:CHANnel:MAIN:SPACing	Sets or queries the main channel spacing.
[SENSe]:MCPower:CHIPrate	Sets or queries the chip rate.
[SENSe]:MCPower:CLEar:RESults	Restarts the average trace.
[SENSe]:MCPower:FREQuency	Sets or queries the center frequency.
[SENSe]:MCPower:FREQuency:STEP	Sets or queries the frequency step size.
[SENSe]:MCPower:FREQuency:STEP:AUTO	Sets or queries whether to set the frequency step size automatically.

Table 2-17: Sense commands (cont.)

Command	Description
[SENSe]:MCPower:NFLoor:STATe	Sets or queries whether to enable the correction for noise floor.
[SENSe]:MCPower:POINts:COUNt	Sets or queries the number of trace points.
[SENSe]:MCPower:RCHannels?	Queries the power reference.
[SENSe]:MCPower:RCHannels:MAIN<x>	Sets the power reference to the main channel with the index (<x>).
[SENSe]:MCPower:RCHannels:TOTal	Sets the power reference to the total power of all the active channels.
[SENSe]:MCPower:RRCRolloff	Sets or queries the filter parameter for the Root Raised Cosine filter.
[SENSe]:OBWidth subgroup	**Occupied Bandwidth measurement**
[SENSe]:OBWidth:AVERage	Sets or queries whether to enable or disable averaging.
[SENSe]:OBWidth:AVERage:COUNt	Sets or queries the number of measurements for averaging.
[SENSe]:OBWidth:{BANDwidth|BWIDth}:MEASurement	Sets or queries the measurement bandwidth.
[SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution]	Sets or queries the resolution bandwidth (RBW).
[SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution]:ACTual?	Queries the actual resolution bandwidth (RBW).
[SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution]:AUTO	Sets or queries whether to set the RBW automatically.
[SENSe]:OBWidth:{BANDwidth|BWIDth}:VIDeo	Sets or queries the video bandwidth (VBW).
[SENSe]:OBWidth:{BANDwidth|BWIDth}:VIDeo:STATe	Sets or queries whether to enable the video bandwidth (VBW).
[SENSe]:OBWidth:CLEar:RESults	Restarts the averaging.
[SENSe]:OBWidth:FREQuency:CENTer	Sets or queries the center frequency.
[SENSe]:OBWidth:FREQuency:STEP	Sets or queries the frequency step size.
[SENSe]:OBWidth:FREQuency:STEP:AUTO	Determines whether to set the frequency step size automatically.
[SENSe]:OBWidth:PERCent	Sets or queries the occupied bandwidth percent power.
[SENSe]:OBWidth:XDBLevel	Sets or queries the x dB level.
[SENSe]:OFDM subgroup	**Orthogonal Frequency Division Multiplexing measurement**
[SENSe]:OFDM:CHANnel:ESTimation	Sets or queries the channel estimation.
[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth]	Sets or queries the channel bandwidth.
[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth]:SRATe?	Returns the FFT sample rate.
[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth]:STATe	Determines whether the channel bandwidth is on or off.
[SENSe]:OFDM:CONSte:DETermination	Sets or queries the constellation determination.
[SENSe]:OFDM:CONSte:DETermination:MANual	Sets or queries a manual constellation type.
[SENSe]:OFDM:FFT:LENGth?	Returns the FFT length.
[SENSe]:OFDM:GUARd:INTerval	Sets or queries the guard interval.
[SENSe]:OFDM:GUARd:INTerval:USER	Sets or queries a user-defined guard interval.
[SENSe]:OFDM:PILot:TRACking:AMPLitude:STATe	Determines whether to select the Amplitude pilot tracking.
[SENSe]:OFDM:PILot:TRACking:PHASe:STATe	Determines whether to select the Phase pilot tracking.
[SENSe]:OFDM:PILot:TRACking:TIMing:STATe	Determines whether to select the Timing pilot tracking.

Table 2-17: Sense commands (cont.)

Command	Description
[SENSe]:OFDM:RADix	Sets the radix for the OFDM measurements.
[SENSe]:OFDM:SCARrier:SPACing	Sets or queries the subcarrier spacing.
[SENSe]:OFDM:SCARrier:SPACing:STATe	Determines whether the subcarrier spacing is on or off.
[SENSe]:OFDM:SCARriers	Sets or queries the Subcarriers to display.
[SENSe]:OFDM:SCARriers:RANGe:STARt	Sets or queries the subcarrier start range.
[SENSe]:OFDM:SCARriers:RANGe:STOP	Sets or queries the subcarrier stop range.
[SENSe]:OFDM:SCARriers:SINGle:INDex	Sets or queries the index of a single subcarrier.
[SENSe]:OFDM:STANdard	Sets or queries the OFDM standard.
[SENSe]:OFDM:SWAP:IQ	Determines whether or not to exchange I and Q data.
[SENSe]:OFDM:SYMBol:ANALysis:OFFSet	Sets or queries the OFDM symbol analysis offset.
[SENSe]:OFDM:SYMBols	Sets or queries the symbols to display.
[SENSe]:OFDM:SYMBols:MAX	Sets or queries the maximum number of symbols to analyze.
[SENSe]:OFDM:SYMBols:MAX:STATe	Enables or disables the maximum number of symbols to analyze.
[SENSe]:OFDM:SYMBols:RANGe:COUNt	Sets or queries the number of symbols to display.
[SENSe]:OFDM:SYMBols:RANGe:STARt	Sets or queries the symbols start range.
[SENSe]:OFDM:SYMBols:SINGle:INDex	Sets or queries the symbol index
[SENSe]:OFDM:UNIT:FREQuency	Specifies the frequency units.
[SENSe]:OFDM:UNIT:TIME	Specifies the time units.
[SENSe]:PHVTime subgroup	**Phase versus Time measurement**
[SENSe]:PHVTime:CLEar:RESults	Restarts multi-trace functions (Average and Max/Min Hold).
[SENSe]:PHVTime:FREQuency:CENTer	Sets or queries the center frequency.
[SENSe]:PHVTime:FREQuency:SPAN	Sets or queries the frequency span.
[SENSe]:PHVTime:FREQuency:STARt	Sets or queries the measurement start frequency.
[SENSe]:PHVTime:FREQuency:STEP	Sets or queries the frequency step size.
[SENSe]:PHVTime:FREQuency:STEP:AUTO	Sets or queries whether to set the frequency step size automatically.
[SENSe]:PHVTime:FREQuency:STOP	Sets or queries the measurement stop frequency.
[SENSe]:PHVTime:MAXTracepoints	Sets or queries the maximum trace points.
[SENSe]:PNOise subgroup (Option 11 only)	**Phase noise measurement**
[SENSe]:PNOise:AVERage:COUNt	Sets or queries the number of traces to combine for averaging.
[SENSe]:PNOise:AVERage:ENABle	Sets or queries whether to enable or disable averaging trace.
[SENSe]:PNOise:CARRier:FREQuency:TRACk	Sets or queries whether to enable tracking the carrier frequency.
[SENSe]:PNOise:CARRier:THReshold	Sets or queries the threshold level to detect the carrier.
[SENSe]:PNOise:CLEar:RESults	Restarts the average process.
[SENSe]:PNOise:FREQuency:INTegration:OFFSet:STARt	Sets or queries the start offset frequency for integration.
[SENSe]:PNOise:FREQuency:INTegration:OFFSet:STOP	Sets or queries the stop offset frequency for integration.

Table 2-17: Sense commands (cont.)

Command	Description
[SENSe]:PNOise:FREQuency:PLOT:OFFSet:STARt	Sets or queries the start offset frequency for plot.
[SENSe]:PNOise:FREQuency:PLOT:OFFSet:STOP	Sets or queries the stop offset frequency for plot.
[SENSe]:PNOise:OPTimization	Sets or queries the method of optimization.
[SENSe]:PULSe subgroup	**Pulsed RF measurements**
[SENSe]:PULSe:ANALyze:FDOMain:MEASurement:TIME:ALENgth	Sets or queries the absolute amount of time included in the Frequency-Domain Linearity measurements.
[SENSe]:PULSe:ANALyze:FDOMain:MEASurement:TIME:METHod	Sets or queries the measurement time method for the Frequency-Domain Linearity measurements.
[SENSe]:PULSe:ANALyze:FDOMain:MEASurement:TIME:RLENgth	Sets or queries the relative amount of time included in the Frequency-Domain Linearity measurements.
[SENSe]:PULSe:ANALyze:FDOMain:MEASurement:TIME:RLEVel	Sets or queries the Frequency-Domain Linearity measurement reference level.
[SENSe]:PULSe:ANALyze:FDOMain:MEASurement:TIME:STARt	Sets or queries the start time for the Frequency-Domain Linearity measurements
[SENSe]:PULSe:ANALyze:IRESponse:CORRection:AMPLitude[:STATe]	Enables or disables the impulse response measurement amplitude corrections for window and for delay of the highest side lobe.
[SENSe]:PULSe:ANALyze:IRESponse:KOTime	Sets or queries the impulse response measurement Keep-out time.
[SENSe]:PULSe:ANALyze:IRESponse:MEASurement:TIME:ALENgth	Sets or queries the absolute amount of time included in impulse response measurements.
[SENSe]:PULSe:ANALyze:IRESponse:MEASurement:TIME:METHod	Sets or queries the impulse response measurement time method.
[SENSe]:PULSe:ANALyze:IRESponse:MEASurement:TIME:RLENgth	Sets or queries the relative amount of time included in Impulse Response measurements.
[SENSe]:PULSe:ANALyze:IRESponse:MEASurement:TIME:RLEVel	Sets or queries the measurement reference level for Impulse Response measurements.
[SENSe]:PULSe:ANALyze:IRESponse:MEASurement:TIME:STARt	Sets or queries the start time for the Impulse Response measurements.
[SENSe]:PULSe:ANALyze:LEVel	Sets or queries how to determine the 50% level.
[SENSe]:PULSe:ANALyze:LEVel:FIFTy	Sets or queries how to determine the 50% level.
[SENSe]:PULSe:ANALyze:LEVel:HUNDred	Sets or queries how to determine the 50% level.
[SENSe]:PULSe:ANALyze:OVERshoot:MEASurement:TIME:ALENgth	Sets or queries the absolute amount of time included in the Overshoot measurements.
[SENSe]:PULSe:ANALyze:OVERshoot:MEASurement:TIME:METHod	Sets or queries the measurement time method for the Overshoot measurements.
[SENSe]:PULSe:ANALyze:OVERshoot:MEASurement:TIME:RLENgth	Sets or queries the relative amount of time included in the Overshoot measurements.
[SENSe]:PULSe:ANALyze:OVERshoot:MEASurement:TIME:RLEVel	Sets or queries the measurement reference level for the Overshoot measurements.
[SENSe]:PULSe:ANALyze:OVERshoot:MEASurement:TIME:STARt	Sets or queries the start time for the Overshoot measurements
[SENSe]:PULSe:ANALyze:PMLocation	Sets or queries the phase measurement location.

Table 2-17: Sense commands (cont.)

Command	Description
[SENSe]:PULSe:ANALyze:POINt:LOCation	Sets or queries the point location method.
[SENSe]:PULSe:ANALyze:RFALl	Sets or queries the threshold levels to measure the rise/fall time.
[SENSe]:PULSe:ANALyze:RIPPle	Sets or queries the ripple portion of the pulse top.
[SENSe]:PULSe:CARRier:FREQuency?	Queries the carrier frequency.
[SENSe]:PULSe:CARRier:OFFSet	Sets or queries the carrier frequency offset.
[SENSe]:PULSe:CARRier:SEARch	Sets or queries how to detect the carrier.
[SENSe]:PULSe:DETect:MEASurement	Sets or queries whether to set the maximum number of pulses.
[SENSe]:PULSe:DETect:NUMBer	Sets or queries the maximum number of pulses to detect.
[SENSe]:PULSe:DETect:POWer[:THReshold]	Sets or queries the power threshold to detect pulses.
[SENSe]:PULSe:DETect:TIMe[:THReshold]	Sets or queries the minimum off-time between pulses.
[SENSe]:PULSe:FILTer:{BANDwidth|BWIDth}	Sets or queries the filter bandwidth for the Gaussian filter.
[SENSe]:PULSe:FILTer:MEASurement	Sets or queries the measurement filter.
[SENSe]:PULSe:FREFerence:AUTO	Determines whether to estimate the frequency reference automatically.
[SENSe]:PULSe:FREFerence:CSLope	Sets or queries the Chirp Slope.
[SENSe]:PULSe:FREFerence:CSLope:AUTO	Sets or queries enabling automatic Chirp Slope estimating.
[SENSe]:PULSe:FREFerence:OFFSet	Sets or queries the frequency reference offset.
[SENSe]:PULSe:MODulation:TYPE	Sets or queries the modulation type.
[SENSe]:PULSe:SIGNal:TYPE	Sets or queries the signal type in the pulsed RF measurements.
[SENSe]:PULSe:STATistics:HISTogram:ORDinate	Sets or queries the Histogram Y unit.
[SENSe]:PULSe:STATistics:HISTogram:POINts	Sets or queries the number of bins in the Histogram plot.
[SENSe]:PULSe:STATistics:HISTogram:POINts:AUTO	Sets or queries enabling automatic Histogram plot bins count.
[SENSe]:REANalyze subgroup	**Spectrogram measurement**
[SENSe]:REANalyze	Sets all measurements to reanalyze the current acquisition record.
[SENSe]:REANalyze:ACQuisition:SETTings?	Queries the settings of all acquisitions in acquisition history.
[SENSe]:REANalyze:ALL	Replay all items in the current acquisition history selection.
[SENSe]:REANalyze:ALL:LOOP	Replay all selected records/frames in a continuous loop.
[SENSe]:REANalyze:CURRent	Replay the currently selected acquisition (or frame).
[SENSe]:REANalyze:CURRent:ACQuisition?	Queries the Acquisition record identifier (ID) of the currently selected acquisition (or frame).
[SENSe]:REANalyze:CURRent:FRAMe?	Queries the Frame identifier (ID) of the currently selected acquisition (or frame).
[SENSe]:REANalyze:CURRent:TIMestamp?	Queries the time stamp of the currently selected acquisition (or frame).
[SENSe]:REANalyze:DELete:DATA	Deletes all acquisition data.
[SENSe]:REANalyze:FIRSt	Replays the first item in the current acquisition history selection.
[SENSe]:REANalyze:LAST	Replays the last item in the current acquisition history selection.
[SENSe]:REANalyze:NEXT	Replays the next item in the current acquisition history selection.
[SENSe]:REANalyze:PAUSe	Sets or queries the reanalysis pause state.

Table 2-17: Sense commands (cont.)

Command	Description
[SENSe]:REANalyze:PREVious	Replays the previous item in the current Replay selection.
[SENSe]:REANalyze:SELect:ACQuisition:DATA:TYPE	Sets or queries the acquisition data type.
[SENSe]:REANalyze:SELect:ACQuisition:FIRSt?	Queries the first selectable acquisition history Start or Stop value.
[SENSe]:REANalyze:SELect:ACQuisition:FIRSt:TIMestamp?	Queries the time stamp associated with the first selectable acquisition history Start or Stop value.
[SENSe]:REANalyze:SELect:ACQuisition:LAST?	Queries the last selectable acquisition history Start or Stop valu
[SENSe]:REANalyze:SELect:ACQuisition:LAST:TIMestamp?	Queries the time stamp associated with the last selectable acquisition history Start or Stop value
[SENSe]:REANalyze:SELect:ACQuisition:STARt	Sets or queries the Acquisition Start value in the acquisition history selection.
[SENSe]:REANalyze:SELect:ACQuisition:STARt:TIMestamp?	Queries the time stamp associated with the current Acquisition Start selection
[SENSe]:REANalyze:SELect:ACQuisition:STOP	Sets or queries the Acquisition Stop value in the acquisition history selection.
[SENSe]:REANalyze:SELect:ACQuisition:STOP:TIMestamp?	Queries the time stamp associated with the current Acquisition Stop selection.
[SENSe]:REANalyze:SELect:ALL	Selects all acquisitions and frames in the current acquisition history.
[SENSe]:REANalyze:SELect:FRAMe:FIRSt?	Queries the first available Frame Start or Stop value available for acquisition history selection.
[SENSe]:REANalyze:SELect:FRAMe:FIRSt:TIMestamp?	Queries the time stamp associated with the first available frame of the first available Acquisition Start or Stop value
[SENSe]:REANalyze:SELect:FRAMe:LAST?	Queries the last available Frame Start or Stop value available for acquisition history selection.
[SENSe]:REANalyze:SELect:FRAMe:LAST:TIMestamp?	Queries the time stamp associated with the last available frame of the last available Acquisition Start or Stop value
[SENSe]:REANalyze:SELect:FRAMe:STARt	Sets or queries the Frame Start value in the acquisition history selection.
[SENSe]:REANalyze:SELect:FRAMe:STARt:TIMestamp?	Queries the time stamp associated with the current Frame Start selection.
[SENSe]:REANalyze:SELect:FRAMe:STOP	Sets or queries the frame stop value in acquisition history selection.
[SENSe]:REANalyze:SELect:FRAMe:STOP:TIMestamp?	Queries the time stamp associated with the current frame stop selection.
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:ALL	Selects all spectra acquisitions and frames in the currently selected set.
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:CURRent:ACQuisition?	Queries the acquisition record identifier (ID) of the currently selected acquisition.
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:CURRent:TIMestamp?	Queries the time stamp of the currently selected acquisition.
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:FIRSt?	Queries the first selectable acquisition start or stop value in the currently selected set.
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:FIRSt:TIMestamp?	Queries the time stamp of the first selectable acquisition history start or stop value in the currently selected acquisition set.

Table 2-17: Sense commands (cont.)

Command	Description
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:LAST?	Queries the last selectable acquisition start or stop value in the currently selected set.
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:LAST:TIMestamp?	Queries the time stamp of the last selectable acquisition start or stop value in the currently selected set.
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STARt	Sets or queries the acquisition start value in the currently selected set.
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STARt:TIMestamp?	Queries the time stamp of the current acquisition start selection.
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STOP	Sets or queries the acquisition stop value in the currently selected set.
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STOP:TIMestamp?	Queries the time stamp of the current acquisition stop selection.
[SENSe]:REANalyze:SPEed	Sets or queries the time delay value inserted between reanalysis of frames or acquisitions.
[SENSe]:REANalyze:STOP	Stops the current reanalysis.
[SENSe]:REANalyze:TIMestamp:DECimal	Sets or queries the number of values to the right of the decimal point to include in all time stamps.
[SENSe]:SEM subgroup	**Spectrum Emissions Mask measurements**
[SENSe]:SEM:{BANDwidth|BWIDth}[:RESolution]	Sets or queries the resolution bandwidth (RBW).
[SENSe]:SEM:{BANDwidth|BWIDth}:VIDeo	Sets or queries the video bandwidth (VBW).
[SENSe]:SEM:{BANDwidth|BWIDth}:VIDeo:STATe	Sets or queries the state of the video bandwidth (VBW) of the reference channel.
[SENSe]:SEM:CHANnel:FILTer	Sets or queries the reference Channel Filter.
[SENSe]:SEM:CHANnel:IBANdwidth	Sets or queries the reference channel integration bandwidth.
[SENSe]:SEM:CHANnel:WIDTh	Sets or queries the reference Channel Filter bandwidth.
[SENSe]:SEM:CHIPrate	Sets or queries the Chip rate.
[SENSe]:SEM:CLEar:RESults	Clears the results in the Spectral Emissions Mask measurement.
[SENSe]:SEM:FREQuency	Sets or queries the measurement frequency.
[SENSe]:SEM:FREQuency:SPAN?	Returns the frequency span.
[SENSe]:SEM:MFILter	Sets or queries the shape of the filter.
[SENSE]:SEM:NFLoor:STATe (This command only applies when an MDO4000B or MDO4000C series instrument is connected.)	Sets or queries if Noise Correction is enabled or disabled.
[SENSE]:SEM:NFLoor:ACQuire (This command only applies when an MDO4000B or MDO4000C series instrument is connected.)	Measures the noise floor on the next acquisition.
[SENSe]:SEM:NFLoor:CORRected? (This command only applies when an MDO4000B or MDO4000C series instrument is connected.)	Queries whether or not noise correction was performed.
[SENSe]:SEM:MODE:REALtime	Sets or queries the Real-time SEM measurement mode.

Table 2-17: Sense commands (cont.)

Command	Description
[SENSe]:SEM:MTYPe	Sets or queries the measurement type.
[SENSe]:SEM:OFFSet:DETection	Sets or queries the detection settings for all offsets.
[SENSe]:SEM:OFFSet<x>:BANDwidth:FILTer?	Returns the bandwidth filter setting.
[SENSe]:SEM:OFFSet<x>:BANDwidth[:RESolution]	Sets or queries the resolution bandwidth.
[SENSe]:SEM:OFFSet<x>:BANDwidth:RESolution:COUNt	Sets or queries the count of Resolution Bandwidths (RBWs).
[SENSe]:SEM:OFFSet<x>:BANDwidth:VIDeo	Sets or queries the video bandwidth setting.
[SENSe]:SEM:OFFSet<x>:BANDwidth:VIDeo:STATe	Sets or queries the state of the video bandwidth.
[SENSe]:SEM:OFFSet<x>:FREQuency:STARt	Sets or queries the start frequency.
[SENSe]:SEM:OFFSet<x>:FREQuency:STOP	Sets or queries the stop frequency.
[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:SAME	Sets or queries the state of the limits for the absolute start and stop settings.
[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:STARt	Sets or queries the absolute power start limit.
[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:STOP	Sets or queries the absolute power stop limit.
[SENSe]:SEM:OFFSet<x>:LIMIT:MASK	Sets or queries the limit mask setting.
[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:SAME	Sets or queries the state of the limits for the relative start and stop settings
[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:STARt	Sets or queries the relative power start limit.
[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:STOP	Sets or queries the relative power stop limit.
[SENSe]:SEM:OFFSet<x>:LIMIT:SIDE	Sets or queries the side of the offset limit.
[SENSe]:SEM:OFFSet<x>:STATe	Determines whether to display the specified row of the Offsets & Limits table.
[SENSe]:SEM:POINts:COUNt	Sets or queries the maximum number of trace points.
[SENSe]:SEM:PRESet	Defines the standard protocol settings for the measurement.
[SENSe]:SEM:REFerence:AUTO	Determines whether to automatically update the Power Reference.
[SENSe]:SEM:REFerence:DETection	Sets or queries the reference channel detection setting.
[SENSe]:SEM:REFerence[:MAGNitude]	Sets or queries the power magnitude.
[SENSe]:SEM:RRCRolloff	Sets or queries the filter parameter (roll-off ratio).
[SENSe]:SEM:STEP:AUTO	Determines whether to automatically update the center frequency.
[SENSe]:SEM:STEP[:MAGNitude]	Sets or queries the step magnitude.
[SENSe]:SEM:WIDTh:DEFinition	Sets or queries the offset definition.
[SENSe]:SGRam subgroup	**Spectrogram measurements**
[SENSe]:SGRam:{BANDwidth|BWIDth}:RESolution	Sets or queries the resolution bandwidth (RBW).
[SENSe]:SGRam:{BANDwidth|BWIDth}[:RESolution]:ACTual?	Queries the actual resolution bandwidth (RBW).
[SENSe]:SGRam:{BANDwidth|BWIDth}[:RESolution]:AUTO	Sets or queries whether to set the RBW automatically.
[SENSe]:SGRam:{BANDwidth|BWIDth}[:RESolution]:MODE	Sets or queries whether to enable or disable the RBW processing.

Table 2-17: Sense commands (cont.)

Command	Description
[SENSe]:SGRam:{BANDwidth|BWIDth}:VIDeo	Sets or queries the video bandwidth (VBW).
[SENSe]:SGRam:{BANDwidth|BWIDth}:VIDeo:STATe	Sets or queries whether to enable the video bandwidth (VBW).
[SENSe]:SGRam:COLor	Sets or queries the color palette of three-dimensional graphs.
[SENSe]:SGRam:COLor:MAXimum	Sets or queries the maximum value of the color axis.
[SENSe]:SGRam:COLor:MINimum	Sets or queries the minimum value of the color axis.
[SENSe]:SGRam:FFT:WINDow	Sets or queries the FFT window.
[SENSe]:SGRam:FILTer[:SHAPe]	Sets or queries the filter shape.
[SENSe]:SGRam:FREQuency:CENTer	Sets or queries the center frequency.
[SENSe]:SGRam:FREQuency:SPAN	Sets or queries the frequency span.
[SENSe]:SGRam:FREQuency:SPAN:BANDwidth[:RESolution]:RATio	Sets or queries the ratio of span to RBW.
[SENSe]:SGRam:FREQuency:SPAN:MAXimum	Sets the frequency range to the maximum real-time span.
[SENSe]:SGRam:FREQuency:STARt	Sets or queries the measurement start frequency.
[SENSe]:SGRam:FREQuency:STEP	Sets or queries the frequency step size.
[SENSe]:SGRam:FREQuency:STEP:AUTO	Sets or queries whether to set the frequency step size automatically.
[SENSe]:SGRam:FREQuency:STOP	Sets or queries the measurement stop frequency.
[SENSe]:SGRam:TIME[:SCALe]:MODe	Sets or queries the vertical mode.
[SENSe]:SGRam:TIME[:SCALe]:OVERlap:PERCent	Sets or queries the Overlap in percent.
[SENSe]:SGRam:TIME[:SCALe]:PER:DIVision	Sets or queries the vertical Time/division value.
[SENSe]:SGRam:TIME[:SCALe]:PER:UPDate:MINutes	Sets or queries the Spectrum Monitor mode Time/Update value for minutes.
[SENSe]:SGRam:WATerfall:ENABle	Sets or queries the Waterfall (3-D spectrogram) display.
[SENSe]:SGRam:TIME[:SCALe]:PER:UPDate:SEConds	Sets or queries the Spectrum Monitor mode Time/Update value
[SENSe]:SGRam:TIME[:SCALe]:SPECtrums:PERLine?	Queries the Spectrums/Line value.
[SENSe]:SGRam:TIME[:SCALe]:STARt:DIVisions	Sets or queries the Vertical offset or Position in divisions.
[SENSe]:SGRam:WATerfall:DIRection	Sets or queries the direction of the Waterfall display.
[SENSe]:SGRam:WATerfall:Y:AUTO	Rescales the height axis automatically to fit the waveform to the screen in the 3-D Waterfall display.
[SENSe]:SGRam:WATerfall:Y:OFFSet	Sets or queries the height offset (the value at the top edge of the vertical axis) of the 3-D Waterfall display.
[SENSe]:SGRam:WATerfall:Y:RESet	Resets the height scale of the 3-D Waterfall display to the default values.
[SENSe]:SGRam:WATerfall:Y[:SCALe]	Sets or queries the height (vertical) range of the Waterfall (3-D spectrogram) display.
[SENSe]:SPECtrum subgroup	**Spectrum measurement**
[SENSe]:SPECtrum:{BANDwidth|BWIDth}[:RESolution]	Sets or queries the resolution bandwidth (RBW).

Table 2-17: Sense commands (cont.)

Command	Description
[SENSe]:SPECtrum:{BANDwidth|BWIDth}[:RESolution]:ACTual?	Queries the actual resolution bandwidth (RBW).
[SENSe]:SPECtrum:{BANDwidth|BWIDth}[:RESolution]:AUTO	Sets or queries whether to set the RBW automatically.
[SENSe]:SPECtrum:{BANDwidth|BWIDth}[:RESolution]:MODE	Sets or queries whether to enable or disable the RBW process.
[SENSe]:SPECtrum:{BANDwidth|BWIDth}:VIDeo	Sets or queries the video bandwidth (VBW).
[SENSe]:SPECtrum:{BANDwidth|BWIDth}:VIDeo:STATe	Sets or queries whether to enable the video bandwidth (VBW).
[SENSe]:SPECtrum:CLEar:RESults	Restarts multi-trace functions (Average and Max/Min Hold).
[SENSe]:SPECtrum:FFT:WINDow	Sets or queries the FFT window.
[SENSe]:SPECtrum:FILTer[:SHAPe]	Sets or queries the filter shape.
[SENSe]:SPECtrum:FREQuency:CENTer	Sets or queries the center frequency.
[SENSe]:SPECtrum:FREQuency:SPAN	Sets or queries the frequency span.
[SENSe]:SPECtrum:FREQuency:SPAN:BANDwidth[:RESolution]:RATio	Sets or queries the ratio of span to RBW.
[SENSe]:SPECtrum:FREQuency:STARt	Sets or queries the measurement start frequency.
[SENSe]:SPECtrum:FREQuency:STEP	Sets or queries the frequency step size.
[SENSe]:SPECtrum:FREQuency:STEP:AUTO	Sets or queries whether to set the frequency step size automatically.
[SENSe]:SPECtrum:FREQuency:STOP	Sets or queries the measurement stop frequency.
[SENSe]:SPECtrum:LENGth	Sets or queries the spectrum length.
[SENSe]:SPECtrum:LENGth:ACTual?	Queries the actual spectrum length.
[SENSe]:SPECtrum:LENGth:AUTO	Sets or queries the spectrum length mode (automatic or manual).
[SENSe]:SPECtrum:MAX:SPAN	Sets the frequency span to the maximum span.
[SENSe]:SPECtrum:POINts:COUNt	Sets or queries the number of sample points on the signal spectrum.
[SENSe]:SPECtrum:STARt	Sets or queries the spectrum offset time.
[SENSe]:SPECtrum:STARt:AUTO	Sets or queries the spectrum offset mode (automatic or manual).
[SENSe]:SPECtrum:TIMe:MODe	Sets or queries the spectrum time parameter mode (automatic or manual).
[SENSe]:SPURious subgroup	**Spurious measurement**
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}	Sets or queries the channel width for the carrier as power reference.
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}:INTegration	Sets or queries the integration bandwidth to calculate the carrier power.
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}[:RESolution]	Sets or queries the resolution bandwidth to measure the carrier power.
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}[:RESolution]:AUTO	Sets or queries whether to set the resolution bandwidth automatically.
[SENSe]:SPURious:CARRier:DETection	Sets or queries the carrier detection method.
[SENSe]:SPURious:CARRier:FREQuency	Sets or queries the carrier frequency.
[SENSe]:SPURious:CARRier:THReshold	Sets or queries the threshold level to detect the carrier.

Table 2-17: Sense commands (cont.)

Command	Description
[SENSe]:SPURious:CLEar:RESults	Restarts multi-trace functions (Average and Max Hold).
[SENSe]:SPURious[:FREQuency]:OVERlap?	Queries whether any of the frequency ranges (A to T) overlap.
[SENSe]:SPURious:LIST	Sets or queries how to list the spurious signals.
[SENSe]:SPURious:MODE	Sets or queries the frequency range mode (Multi or Single).
[SENSe]:SPURious:OPTimization	Sets or queries the method of optimization.
[SENSe]:SPURious:POINts:COUNt	Sets or queries the trace point count for the frequency range.
[SENSe]:SPURious:RANGe<x>:BANDwidth:VIDeo	Sets or queries the VBW for the specified frequency range.
[SENSe]:SPURious:RANGe<x>:BANDwidth:VIDeo:STATe	Sets or queries whether to enable the VBW for the frequency range.
[SENSe]:SPURious:RANGe<x>:DETection	Sets or queries the display detector for the frequency range.
[SENSe]:SPURious:RANGe<x>:EXCursion	Sets or queries the excursion level in the frequency range.
[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]	Sets or queries the filter shape for the frequency range.
[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]:BANDwidth	Sets or queries the filter bandwidth for the frequency range.
[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]:BANDwidth:AUTO	Sets or queries whether to set the filter bandwidth automatically.
[SENSe]:SPURious:RANGe<x>:FREQuency:STARt	Sets or queries the start frequency of the range.
[SENSe]:SPURious:RANGe<x>:FREQuency:STOP	Sets or queries the stop frequency of the range.
[SENSe]:SPURious:RANGe<x>:LIMit:ABSolute:STARt	Sets or queries the absolute start amplitude of the limits for the range.
[SENSe]:SPURious:RANGe<x>:LIMit:ABSolute:STOP	Sets or queries the absolute stop amplitude of the limits for the range.
[SENSe]:SPURious:RANGe<x>:LIMit:MASK	Sets or queries the limit mask function mode for the frequency range.
[SENSe]:SPURious:RANGe<x>:LIMit:RELative:STARt	Sets or queries the relative start amplitude of the limits for the range.
[SENSe]:SPURious:RANGe<x>:LIMit:RELative:STOP	Sets or queries the relative stop amplitude of the limits for the range.
[SENSe]:SPURious:RANGe<x>:STATe	Sets or queries whether to enable or disable the frequency range.
[SENSe]:SPURious:RANGe<x>:THReshold	Sets or queries the threshold level to detect spurious signals in a range.
[SENSe]:SPURious:REFerence	Sets or queries the power reference.
[SENSe]:SPURious:REFerence:MANual:POWer	Sets or queries the reference power level.
[SENSe]:SQUality subgroup	**Signal quality**
[SENSe]:SQUality:EVM:ALL[:STATe]	Sets or queries the display of Offset EVM results.
[SENSe]:TOVerview subgroup	**Time overview**
[SENSe]:TOVerview:FREQuency:CENTer	Sets or queries the center frequency.
[SENSe]:TOVerview:MAXTracepoints	Sets or queries the maximum trace points.
[SENSe]:TOVerview:TIME:PER:DIVision	Sets or queries the time per division (dB/div) value in the Time Overview display.
[SENSe]:USETtings subgroup	**All measurements**
[SENSe]:USETtings	Updates the analyzer settings.
[SENSe}:WLAN subgroup	**Wireless LAN settings**
[SENSe]:WLAN:ANALysis:LENGth	Specifies the length of the analysis period.

Table 2-17: Sense commands (cont.)

[SENSe]:WLAN:ANALysis:LENGth:ACTual?	Returns the value of the actual analysis length.
[SENSe]:WLAN:ANALysis:LENGth:AUTO	Sets the analysis length value based on the requirements of the selected display.
[SENSe]:WLAN:ANALysis:OFFSet	Specifies or queries the analysis offset in the symbol interval.
[SENSe]:WLAN:BURSt	Specifies the burst to measure when multiple bursts are present in an acquisition.
[SENSe]:WLAN:CHANnel[:BANDwidth|:BWIDth]	Specifies or queries a nominal channel bandwidth to use for the WLAN measurements.
[SENSe]:WLAN:EQUalizer:TRAining	Specifies the method for estimating channel frequency response and equalization.
[SENSe]:WLAN:FFT:LENGth?	Returns the fixed FFT length for the WLAN measurements.
[SENSe]:WLAN:FFT:SRATe?	Returns the FFT sample rate for the WLAN measurements.
[SENSe]:WLAN:FREQuency:ERRor	Sets or queries the frequency error in the WLAN analysis parameters.
[SENSe]:WLAN:FREQuency:ERRor:AUTO	Turns on the automatic determination of the frequency error.
[SENSe]:WLAN:GUARd:INTerval	Sets or queries the Guard Interval to use in the input signal.
[SENSe]:WLAN:MODulation:MANual	Specifies a fixed modulation type.
[SENSe]:WLAN:MODulation:SOURce	Specifies the method of data symbol modulation identification.
[SENSe]:WLAN:PILot:TRACking:AMPLitude:STATe	Specifies the pilot subcarrier to use to correct amplitude variations over the packet.
[SENSe]:WLAN:PILot:TRACking:PHASe:STATe	Specifies the pilot subcarrier to use to correct phase variations over the packet.
[SENSe]:WLAN:PILot:TRACking:TIMing:STATe	Specifies the pilot subcarrier to use to correct timing variations over the packet.
[SENSe]:WLAN:RADix	Specifies or queries the format for displaying the symbol values.
[SENSe]:WLAN:SCARriers	Sets or queries which subcarriers to display in the WLAN view.
[SENSe]:WLAN:SCARriers:RANGe:STARt	Sets or queries the start value of the subcarrier range.
[SENSe]:WLAN:SCARriers:RANGe:STOP	Sets or queries the stop value of the subcarrier range.
[SENSe]:WLAN:SCARriers:SINGle:INDex	Specifies or queries the single subcarrier to display in the WLAN view.
[SENSe]:WLAN:SCARriers:SPACing	Sets or queries the subcarrier spacing.
[SENSe]:WLAN:SCARriers:SPACing:AUTO	Automatically sets the subcarrier spacing.
[SENSe]:WLAN:SIGNal:STANdard:TYPE	Sets or queries the WLAN signal standard.
[SENSe]:WLAN:SubCARrier:DERotation	Shows subcarriers for displays with or without the Gamma subcarrier phase rotation removed.
[SENSe]:WLAN:SUMMary:CLEar:RESults	Clears the results of the WLAN Summary display.
[SENSe]:WLAN:SUMMary:CLEar:RESults	Clears the results in the WLAN Summary measurement.
[SENSe]:WLAN:SUMMary:EVM:RMS:AVERage:COUNt	Specifies the maximum number of bursts to average.
[SENSe]:WLAN:SUMMary:EVM:RMS:AVERage:COUNt:ENABle	Enables the burst average function.
[SENSe]:WLAN:SWAP:IQ	Specifies to swap the I and Q components of a signal.

Table 2-17: Sense commands (cont.)

[SENSe]:WLAN:SYMBols|CHIPs	Specifies or queries the symbols or chips are displayed in the WLAN Symbol Table.
[SENSe]:WLAN:SYMBols|CHIPs:MAX	Sets or queries the value for the maximum number of symbols/chips to analyze.
[SENSe]:WLAN:SYMBols|CHIPs:MAX:STATe	Turns the maximum data symbols to analyze feature on or off.
[SENSe]:WLAN:SYMBols|CHIPs:RANGe:COUNt	Sets or queries the number of symbols or chips to display.
[SENSe]:WLAN:SYMBols|CHIPs:RANGe:STARt	Sets or queries the symbols/chips start value.
[SENSe]:WLAN:SYMBols|CHIPs:RANGe:STOP	Sets or queries the symbols or chips stop value.
[SENSe]:WLAN:SYMBols|CHIPs:SINGle:INDex	Sets or queries the symbol or chip to use when displaying results from a single symbol.
[SENSe]:WLAN:UNIT:FREQuency	Specifies or queries whether the displayed frequency units are frequency (Hz) or subcarrier.
[SENSe]:WLAN:UNIT:TIME	Specifies or queries whether the displayed time units are seconds, symbols or chips.
SENSe:LTE subgroup	
SENSe:LTE:ACLR:BANDwidth:RESolution	Sets or queries the resolution bandwidth (RBW).
SENSe:LTE:ACLR:BANDwidth:VIDeo	Sets or queries the video bandwidth (VBW).
SENSe:LTE:ACLR:BANDwidth:VIDeo:STATe	Sets or queries enabling or disabling the video bandwidth (VBW).
SENSe:LTE:ACLR:CHANnel:IBANdwidth	Sets or queries the reference channel integration bandwidth.
SENSe:LTE:ACLR:CHANnel:WIDTh	Sets or queries the reference channel width.
SENSe:LTE:ACLR:CLEar:RESults	Clears the results in the measurement.
SENSe:LTE:ACLR:FREQuency	Sets or queries the measurement center frequency.
SENSe:LTE:ACLR:MFILter	Sets or queries the shape of the filter.
SENSe:LTE:ACLR:MODE:REALtime	Sets or queries the real-time LTE ACLR measurement mode.
SENSe:LTE:ACLR:NFLoor:ACQuire	Enables measurement of the noise floor.
SENSe:LTE:ACLR:NFLoor:CORRected?	Returns whether or not noise correction was applied on the last acquisition.
SENSe:LTE:ACLR:NFLoor:STATe	Determines whether to enable or disable the correction for noise floor.
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution	Sets or queries the resolution bandwidth (RBW) value in the specified row in the Offsets & Limits table.
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:COUNt	Sets or queries the count for the resolution bandwidths in the specified row in the Offsets & Limits table.
SENSe:LTE:ACLR:OFFSet<X>:BANDwidth:RESolution:FILTer	Sets or queries the filter bandwidth resolution under Offset & Limit table.
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:VIDeo	Sets or queries the resolution video bandwidth (VBW) value in the specified row in the Offsets & Limits table.
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:VIDeo:STATe	Sets or queries whether to enable or disable the video resolution bandwidth (VBW) in the specified row in the Offsets & Limits table.
SENSe:LTE:ACLR:OFFSet<x>:DETection	Sets or queries the detection settings for all offsets.
SENSe:LTE:ACLR:OFFSet<x>:FREQuency:BANDwidth	Sets or queries the frequency bandwidth value for the specified row of the Offsets & Limits table.

Table 2-17: Sense commands (cont.)

SENSe:LTE:ACLR:OFFSet<x>:FREQuency:OFFSet	Sets or queries the frequency offset value of the specified row in the Offsets & Limits table.
SENSe:LTE:ACLR:OFFSet<x>:LIMit:ABSolute	Sets or queries the absolute limit value of the row in the Offset & Limits table.
SENSe:LTE:ACLR:OFFSet<x>:LIMit:MASK	Sets or queries the limit mask setting for the specified row in the Offsets & Limits table.
SENSe:LTE:ACLR:OFFSet<x>:LIMit:RELative	Sets or queries the relative limit value of channel under Offset & Limits table.
SENSe:LTE:ACLR:OFFSet<x>:LIMit:SIDE	Sets or queries the side of the offset limit for the specified row in the Offsets & Limits table.
SENSe:LTE:ACLR:OFFSet<x>:STATe	Sets or queries the specified row of the Offsets & Limits table.
SENSe:LTE:ACLR:POINts:COUNt	Sets or queries the number of points used for the trace.
SENSe:LTE:ACLR:REFerence:MAGNitude:AUTO	Sets or queries the reference magnitude to automatic or manual.
SENSe:LTE:ACLR:REFerence:MAGNitude:DETection	Sets or queries the reference magnitude detection.
SENSe:LTE:ACLR:REFerence:MAGNitude	Sets or queries the reference power magnitude.
SENSe:LTE:ACLR:STEP:AUTO	Determines whether to automatically update the center frequency increment/decrement step size.
SENSe:LTE:ACLR:STEP:MAGNitude	Sets or queries the step magnitude for the center frequency.
SENSe]:LTE:ANALysis:EQUAlization:STATe	Sets or queries the Enable Equalization state in the LTE Analysis Params tab.
SENSe:LTE:ANALysis:LENGth:ACTual?	Queries the actual analysis length for the LTE Constellation and LTE Power vs Time displays.
[SENSe]:LTE:ANALysis:OFFSet	Sets or queries the LTE analysis offset for the LTE Constellation or LTE Power vs Time.
SENSe:LTE:ANALysis:OFFSet:AUTO	Sets or queries the Auto checkbox state for analysis offset.
SENSe:LTE:ANALysis:TIMe:UNIT	Sets or queries the unit preference under the LTE Analysis tab.
SENSe:LTE:CHANnel:BANDwidth	Specifies or queries the channel bandwidth to use.
SENSe:LTE:CHSPectrum:AVERage	Sets or queries how to average the waveform.
SENSe:LTE:CHSPectrum:AVERage:COUNt	Sets or queries the number of traces for averaging.
SENSe:LTE:CHSPectrum:BANDwidth:RESolution	Sets or queries the resolution bandwidth (RBW).
SENSe:LTE:CHSPectrum:BANDwidth:RESOlution:AUTO	Determines whether to set the resolution bandwidth (RBW) automatically or manually. Queries this setting.
SENSe:LTE:CHSPectrum:BANDwidth:VIDeo	Sets or queries the video bandwidth (VBW).
SENSe:LTE:CHSPectrum:BANDwidth:VIDeo:STATe	Sets or queries enabling or disabling the video bandwidth (VBW).
SENSe:LTE:CHSPectrum:CHANnel:BANDwidth	Specifies or queries a nominal channel bandwidth to use.
SENSe:LTE:CHSPectrum:CLEar:RESults	Clears the results in the LTE Channel Spectrum measurement.
SENSe:LTE:CHSPectrum:DETection	Sets or queries the trace detection settings.
SENSe:LTE:CHSPectrum:FREQuency	Sets or queries the measurement center frequency.
SENSe:LTE:CHSPectrum:FREQuency:SPAN	Sets or queries the frequency span.
SENSe:LTE:CHSPectrum:FREQuency:STEP	Sets or queries the frequency step size.

Table 2-17: Sense commands (cont.)

SENSe:LTE:CHSPectrum:FREQuency:STEP:AUTO	Determines whether to set the frequency step size automatically or manually. Queries this setting.
SENSe:LTE:CHSPectrum:POINts:COUNt	Sets or queries the number of points used for the trace.
SENSe:LTE:FRAMe:STRUcture	Sets or queries the frame structure.
SENSe:TOFF:POWEr:SPECtral:DENSity	Sets or queries the limit set for comparison of the Toff power spectral density value.

System Commands

Use the SYSTem commands to set or query system parameters for operation.

Table 2-18: System commands

Header	Description
SYSTem:PRESet:BLUEtooth:STANdard	Sets or queries the standard, setup, Retaining Current Reflevel, and Retaining Center Frequency in the Bluetooth standard preset.
SYSTem:PRESet:P25:STANdard	Restores the analyzer defaults for the P25 Standards preset.
SYSTem:SVPC:CONNect	Sets or queries an instrument connection to SignalVu-PC.
SYSTem:SVPC:CONNect: RSA300	Connects an RSA306 instrument to SignalVu-PC.
SYSTem:DATE	Sets or queries the current date.
SYSTEM:SVPC:DISConnect	Disconnects an instrument.
SYSTEM:SVPC:DEVice:WRITe (Only applies when an MDO4000B series or MDO4000C series instrument is connected.)	Write command (if the query is true).
SYSTEM:SVPC:DEVice:READ (Only applies when an MDO4000B series or MDO4000C series instrument is connected.)	Read command (if the query is true).
SYSTEM:SVPC:DEVice:QUERy?	Queries if the acquisition engine supports PI Control (for instance, the MDO4000C does, performance oscilloscopes do not).
SYSTem:ERRor:ALL?	Queries all the error or event information.
SYSTem:ERRor:CODE:ALL?	Queries all the error or event codes.
SYSTem:ERRor:CODE[:NEXT]?	Queries the latest error or event information.
SYSTem:ERRor:COUNt?	Queries the number of errors or events.
SYSTem:ERRor[:NEXT]?	Queries the latest error or event information.
SYSTem:SVPC:KLOCk (Only applies when an MDO4000B series or MDO4000C series instrument is connected.)	Enables or disables the front panel lockout operation and queries lock status.
SYSTem:OPTions?	Queries optional information.
SYSTem:PRESet	Restores the analyzer to the defaults.
SYSTem:PRESet:APPLication	Restores the analyzer to the defaults for the application preset type.
SYSTem:PRESet:APPLication:ACTion	Sets or queries the preset action for the application preset type.
SYSTem:PRESet:APPLication:SELected	Sets or queries the presets for the application preset type.
SYSTem:PRESet:DPX	Presets the analyzer.
SYSTem:PRESet:DPX:ACTion	Sets or queries the preset action for the DPX preset type.
SYSTem:PRESet:DPX:SELected	Sets or queries the presets for the DPX preset type.
SYSTem:PRESet[:MAIN]	Restores the analyzer to the defaults.
SYSTem:PRESet:MAIN:ACTion	Sets or queries the preset action for the main preset type.
SYSTem:PRESet:MAIN:SELected	Sets or queries the presets for the main preset type.

Table 2-18: System commands (cont.)

Header	Description
SYSTem:PRESet:STANdards	Restores the instrument to the defaults for the WLAN standards preset type.
SYSTem:PRESet:STANdards:ACTion	Sets or queries the preset action for the standards preset type.
SYSTem:PRESet:USER	Restores the analyzer to the defaults for the user preset type.
SYSTem:PRESet:USER:ACTion	Sets or queries the preset action for the user preset type.
SYSTem:PRESet:USER:SELected	Sets or queries the presets for the user preset type.
SYSTem:PRESet:WLAN:STANdard	Sets or queries the standard and bandwidth presets for the WLAN standards preset type.
SYSTem:TIME	Sets or queries the current time.
SYSTem:VERSion?	Queries the version of the SCPI.
SYSTem:SVPC:INSTrument:SERial?	Queries the serial number and model of the connected instrument. This commands will not return the correct instrument serial number if an RSA is disconnected and then another connected without using the Search for Instruments tool in the SignalVu-PC application to find and connect to the new instrument.
SYSTem:SVPC:INSTrument:VERSion?	Queries the firmware version of the connected instrument. This commands will not return the correct instrument firmware version if an RSA is disconnected and then another connected without using the Search for Instruments tool in the SignalVu-PC application to find and connect to the new instrument.
SYSTem:SVPC:INSTrument:MODEL?	Queries the model of the connected instrument. This commands will not return the correct instrument model if an RSA is disconnected and then another connected without using the Search for Instruments tool in the SignalVu-PC application to find and connect to the new instrument.
SYSTem:SVPC:INSTrument:OPTions?	Queries the options for the connected instrument. This commands will not return the correct instrument options if an RSA is disconnected and then another connected without using the Search for Instruments tool in the SignalVu-PC application to find and connect to the new instrument.
SYSTem:LTE subgroup	
SYSTem:PRESet:LTE:STANdard	Sets or queries LTE standard preset parameters.

Trace Commands

Use the TRACe commands to select trace type and to control trace arithmetic.

Table 2-19: Trace commands

Header	Description
TRACe<x>:{AM|FM|PM} subgroup	**AM/FM/PM measurement**
TRACe<x>:{AM|FM|PM}	Determines whether or not to show the specified trace.
TRACe:{AM|FM|PM}:DETection	Sets or queries the display detector, the method to be used for decimating traces to fit the available horizontal space on screen.
TRACe:{AM|FM|PM}:FREeze	Sets or queries whether or not to freeze the trace display.
TRACe:{AM|FM|PM}:FUNCtion	Sets or queries the trace function.
TRACe<x>:Bluetooth subgroup	**Bluetooth measurement**
TRACe<x>:BLUEtooth:CONSte	Determines whether or not to show the specified trace.
TRACe<x>:BLUEtooth:CONSte:FREeze	Sets or queries whether or not to freeze the specified trace.
TRACe<x>:BLUEtooth:CONSte:MODE	Sets or queries the trace display mode.
TRACe<x>:BLUEtooth:CONSte:SELect	Selects the specified trace.
TRACe<x>:BLUEtooth:EDIagram:ENABle:I	Determines whether to show or hide the I trace.
TRACe<x>:BLUEtooth:EDIagram:ENABle:Q	Determines whether to show or hide the Q trace.
TRACe<x>:BLUEtooth:EDIagram:SELect:I	Selects the I trace or queries if the I trace is selected or not.
TRACe<x>:BLUEtooth:EDIagram:SELect:Q	Selects the Q trace or queries if the Q trace is selected or not.
TRACe<x>:BLUEtooth:FDVTime:MODE	Sets or queries whether to display the Frequency Deviation vs. Time trace as vectors or symbols (points).
TRACe:P25 subgroup	**APCO P25 measurement**
TRACe<x>:P25:CONStellation	Determines whether or not to show the specified trace.
TRACe<x>:P25:CONStellation:FREeze	Determines whether or not to freeze (halt updates of) the specified trace.
TRACe<x>:P25:CONStellation:MODE	Sets or queries how to display the trace content.
TRACe<x>:P25:CONStellation:SELect	Selects the specified trace.
TRACe<x>:P25:EDIagram:ENABle:I	Determines whether to show or hide the I trace.
TRACe<x>:P25:EDIagram:ENABle:Q	Determines whether to show or hide the Q trace.
TRACe<x>:P25:EDIagram:SELect:I	Selects the I trace or queries if the I trace is selected or not.
TRACe<x>:P25:EDIagram:SELect:Q	Selects the Q trace or queries if the Q trace is selected or not.

Table 2-19: Trace commands (cont.)

Header	Description
TRACe:Noise subgroup	**Noise Figure and Gain measurement**
TRACe<x>:NOISe:FIGure:AVERageCOUNt	Sets or queries the average counts for the specified trace.
TRACe<x>:NOISe:FIGure:COUNt:ENABle	Sets to enable the count feature, or queries the enables status, of the specified trace.
TRACe<x>:NOISe:FIGure:DETection	Sets or queries the trace detection method.
TRACe<x>:NOISe:FIGure:FREeze	Determines whether or not to freeze (halt updates of) the specified trace.
TRACe<x>:NOISe:FIgure:FUNCtion	Sets or queries the trace processing method (function) in the display.
TRACe<x>:NOISe:FIGure:SELect	Selects the specified trace.
TRACe<x>:NOISe:FIGure:SHOW	Determines whether or not to show the specified trace.
TRACe<x>:NOISe:GAIN:AVERage:COUNt	Sets or queries the average counts for the specified trace.
TRACe<x>:NOISe:GAIN:COUNt:ENABle	Enables or disables the count feature for the specified trace.
TRACe<x>:NOISe:GAIN:DETection	Sets or queries the trace detection method.
TRACe<x>:NOISe:GAIN:FREeze	Determines whether or not to freeze (halt updates of) the specified trace.
TRACe<x>:NOISe:GAIN:FUNCtion	Sets or queries the trace processing method (function) in the display.
TRACe<x>:NOISe:GAIN:SELect	Selects the specified trace.
TRACe<x>:NOISe:GAIN:SHOW	Determines whether or not to show the specified trace.
TRACe<x>:NOISe:TEMPerature:AVERageCOUNt	Sets or queries the average counts for the specified trace.
TRACe<x>:NOISe:TEMPerature:COUNt:ENABle	Enables or disables the count feature for the specified trace.
TRACe<x>:NOISe:TEMPerature:DETection	Sets or queries the trace detection method.
TRACe<x>:NOISE:TEMPerature:FREeze	Determines whether or not to freeze (halt updates of) the specified trace.
TRACe<x>:NOISe:TEMPerature:FUNCtion	Sets or queries the trace processing method (function) in the display.
TRACe<x>:NOISe:TEMPerature:SELect	Selects the specified trace.
TRACe<x>:NOISe:TEMPerature:SHOW	Determines whether or not to show the specified trace.
TRACe<x>:NOISe:YFACtor:AVERage:COUNt	Sets or queries the average counts for the specified trace.
TRACe<x>:NOISe:YFACtor:COUNt:ENABle	Enables or disables the count feature for the specified trace.
TRACe<x>:NOISe:YFACtor:DETection	Sets or queries the trace detection method.
TRACe<x>:NOISE:YFACtor:FREeze	Determines whether or not to freeze (halt updates of) the specified trace.
TRACe<x>:NOISe:YFACtor:FUNCtion	Sets or queries the trace processing method (function) in the display.
TRACe<x>:NOISe:YFACtor:SELect	Selects the specified trace.
TRACe<x>:NOISe:YFACtor:SHOW	Determines whether or not to show the specified trace.
TRACe<x>:AVTime subgroup	**Amplitude versus time measurement**
TRACe<x>:AVTime	Sets or queries whether or not to show the specified trace.
TRACe:{FSETtling|PSETtling}:AVERage:COUNt	Sets or queries the number of traces to combine for averaging.
TRACe<x>:AVTime:AVERage:COUNt	Sets or queries the number of traces to combine.
TRACe<x>:AVTime:AVERage:RESet	Clears the average data and resets the average counter.
TRACe<x>:AVTime:COUNt	Sets or queries the count for the Max or Min Hold trace.
TRACe<x>:AVTime:COUNt:ENABle	Sets or queries whether or not to enable the count for Max/Min Hold.

Table 2-19: Trace commands (cont.)

Header	Description
TRACe<x>:AVTime:COUNt:RESet	Clears the Max or Min Hold data and counter, and restarts the process.
TRACe<x>:AVTime:DETection	Sets or queries the display detector.
TRACe<x>:AVTime:FREeze	Sets or queries whether or not to freeze the display of the trace.
TRACe<x>:AVTime:FUNCtion	Sets or queries the trace function.
TRACe<x>:AVTime:LEFToperand	Sets or queries the left operand for the math trace.
TRACe<x>:AVTime:RIGHtoperand	Sets or queries the right operand for the math trace.
TRACe<x>:AVTime:SELect	Sets or queries the trace number to display the readout.
TRACe<x>:CCDF subgroup	**CCDF measurement**
TRACe<x>:CCDF:FREeze	Sets or queries whether or not to freeze the display of the trace.
TRACe<x>:CCDF:SELect	Sets or queries the trace number to display the readout.
TRACe<x>:CCDF:SHOW	Sets or queries whether to show or hide the trace.
TRACe<x>:CCDF:X	Sets or queries the horizontal position of the measurement pointer.
TRACe<x>:CCDF:Y?	Queries the vertical position (CCDF value) of the measurement pointer.
TRACe:CONSte subgroup (Option 21 only)	**Constellation measurement**
TRACe<x>:CONSte	Displays or hides the specified trace in the Constellation display.
TRACe<x>:CONSte:FREeze	Halts updates to the specified trace in the Constellation display.
TRACe<x>:CONSte:MODE	Sets or queries how to display the Constellation trace.
TRACe<x>:CONSte:Q:OFFSet	Sets the Q offset in the Constellation display to Use shared Pref, Remove Q offset, or Include Q offset.
TRACe<x>:CONSte:SELect	Select the trace in the Constellation display to which settings are applied.
TRACe:DIQVtime subgroup (Option 21 only)	**Demodulated I&Q versus time measurement**
TRACe:DIQVtime:ENABle:I	Sets or queries whether to show or hide the trace I.
TRACe:DIQVtime:ENABle:Q	Determines whether to show or hide the Q trace.
TRACe:DIQVtime:MODE	Sets or queries whether to display the Demod I&Q vs Time trace as vectors or symbols.
TRACe:DIQVtime:SELect:I	Selects the I trace.
TRACe:DIQVtime:SELect:Q	Selects the Q trace.
TRACe<x>:DPX subgroup	**DPX spectrum measurement**
TRACe<x>:DPX	Sets or queries whether or not to show the waveform.
TRACe<x>:DPX:AVERage:COUNt	Sets or queries the number of traces to combine for averaging.
TRACe<x>:DPX:COLor:CURVe	Sets or queries the color intensity.
TRACe<x>:DPX:COLor:INTensity	Sets or queries the color intensity in the DPX spectrum view.
TRACe<x>:DPX:COLor:SCALe:AUTO	Automatically adjusts the Max and Min color settings to display the broadest range of colors.
TRACe<x>:DPX:DETection	Sets or queries the function.
TRACe<x>:DPX:DGRAM:SELect:LINE	Sets or queries a line in the DPXogram display to send to the DPX spectrum display.
TRACe<x>:DPX:DOT:PERSistent	Sets or queries whether to enable or disable the dot persistence.

Table 2-19: Trace commands (cont.)

Header	Description
TRACe<x>:DPX:DOT:PERSistent:TYPE	Sets or queries the persistence type.
TRACe<x>:DPX:DOT:PERSistent:VARiable	Sets or queries the length of time that data points are displayed.
TRACe<x>:DPX:FREeze	Sets or queries whether or not to freeze the display of the trace.
TRACe<x>:DPX:FUNCtion	Sets or queries the trace function for the in the DPX spectrum view.
TRACe<x>:DPX:LEFToperand	Sets or queries the left operand for the math trace.
TRACe<x>:DPX:RIGHtoperand	Sets or queries the right operand for the math trace.
TRACe<x>:DPX:SELect	Sets or queries the trace number to display the readout.
TRACe:EDIagram subgroup (Option 21 only)	**Eye diagram**
TRACe:EDIagram:ENABle:I	Sets or queries whether to show or hide the trace I.
TRACe:EDIagram:ENABle:Q	Sets or queries whether to show or hide the trace Q.
TRACe:EDIagram:SELect:I	Selects the I trace.
TRACe:EDIagram:SELect:Q	Selects the Q trace.
TRACe:EVM subgroup	**EVM measurement**
TRACe:EVM:MODE	Sets or queries whether to display the EVM vs Time trace as vectors or symbols.
TRACe:FDVTime subgroup	**Frequency deviation versus time measurement**
TRACe:FDVTime:MODE	Sets or queries whether to display the Frequency Deviation vs Time trace as vectors or symbols.
TRACe:FVTime subgroup	**Frequency versus time measurement**
TRACe:FVTime	Sets or queries whether or not to show the trace.
TRACe:FVTime:AVERage:COUNt	Sets or queries the number of traces to combine for averaging.
TRACe:FVTime:COUNt	Sets or queries the count for the Max or Min Hold trace.
TRACe:FVTime:COUNt:ENABle	Sets or queries whether or not to enable the count for Max/Min Hold.
TRACe:FVTime:COUNt:RESet	Clears the Max or Min Hold data and counter, and restarts the process.
TRACe<x>:FVTime:DETection	Enables or queries the type of detection for the specified trace.
TRACe:FVTime:FREeze	Sets or queries whether or not to freeze the display of the trace.
TRACe:FVTime:FUNCtion	Sets or queries the trace function.
TRACe:IQVTime subgroup	**RF I&Q versus time measurement**
TRACe:IQVTime:AVERage:COUNt	Sets or queries the number of traces to combine for averaging.
TRACe:IQVTime:COUNt	Sets or queries the count for the Max or Min Hold trace.
TRACe:IQVTime:COUNt:ENABle	Determines whether to enable or disable the count for the Max or Min Hold trace.
TRACe:IQVTime:DETection	Sets or queries whether or not to enable the count for Max/Min Hold.
TRACe:IQVTime:COUNt:RESet	Clears the Max or Min Hold data and counter, and restarts the process.
TRACe:IQVTime:ENABle:I	Sets or queries whether to show or hide the trace I.
TRACe:IQVTime:ENABle:Q	Sets or queries whether to show or hide the trace Q.
TRACe:IQVTime:FREeze	Sets or queries whether to freeze the IQ traces.

Table 2-19: Trace commands (cont.)

Header	Description
TRACe:IQVTime:FUNCtion	Sets or queries the trace function.
TRACe:IQVTime:SELect:I	Sets or queries whether to choose the I trace.
TRACe:IQVTime:SELect:Q	Sets or queries whether to choose the Q trace.
TRACe:MERRor subgroup	**Magnitude Error measurement**
TRACe:MERRor:MODE	Sets or queries whether to display the Magnitude error trace as vectors or symbols.
TRACe:OBWidth subgroup	**Occupied Bandwidth measurement**
TRACe:OBW:MAXHold	Determines whether to enable or disable the Max Hold trace.
TRACe:OFDM subgroup	**OFDM measurements**
TRACe:OFDM:CONSte	Determines whether or not to show the trace.
TRACe:OFDM:CONSte:FREeze	Determines whether or not to freeze the display of the trace.
TRACe<x>:OFDM:EVM	Determines whether or not to show the specified trace.
TRACe<x>:OFDM:EVM:FREeze	Determines whether or not to freeze the display of the specified trace.
TRACe<x>:OFDM:EVM:SELect	Sets or queries the specified trace to display.
TRACe<x>:OFDM:MERRor	Determines whether or not to show the specified trace.
TRACe<x>:OFDM:MERRor:FREeze	Determines whether or not to freeze the display of the specified trace.
TRACe<x>:OFDM:MERRor:SELect	Sets or queries the specified trace to display.
TRACe<x>:OFDM:PERRor	Determines whether or not to show the specified trace.
TRACe<x>:OFDM:PERRor:FREeze	Determines whether or not to freeze the display of the specified trace.
TRACe<x>:OFDM:PERRor:SELect	Sets or queries the specified trace to display.
TRACe<x>:OFDM:POWer	Determines whether or not to show the specified trace.
TRACe<x>:OFDM:POWer:FREeze	Determines whether or not to freeze the display of the specified trace.
TRACe<x>:OFDM:POWer:SELect	Sets or queries the specified trace to display.
TRACe:OFDM:FLATness:FREeze	Sets or queries whether to freeze the average trace
TRACe:OFDM:FLATness:SHOW	Displays or hides the trace on the OFDM Spectral Flatness display.
TRACe:PERRpr subgroup	**Phase Error measurement**
TRACe:PERRor:MODE	Sets or queries whether to display the Frequency Deviation vs Time trace as vectors or symbols.
TRACe:PHVTime subgroup	**Phase versus time measurement**
TRACe:PHVTime	Sets or queries whether or not to show the trace.
TRACe:PHVTime:AVERage:COUNt	Sets or queries the number of traces to combine for averaging.
TRACe:PHVTime:COUNt	Sets or queries the count for the Max or Min Hold trace.
TRACe:PHVTime:COUNt:ENABle	Sets or queries whether or not to enable the count for Max/Min Hold.
TRACe:PHVTime:COUNt:RESet	Clears the Max or Min Hold data and counter, and restarts the process.
TRACe<x>:PHVTime:DETection	Sets or queries the display detector.
TRACe<x>:{FSETtling|PSETtling}:FREeze	Sets or queries whether to freeze the trace.

Table 2-19: Trace commands (cont.)

Header	Description
TRACe:PHVTime:FREeze	Determines whether to freeze the trace display in the Phase versus Time measurement.
TRACe:PHVTime:FUNCtion	Sets or queries the trace function.
TRACe:PNOise subgroup (Option 11 only)	**Phase noise measurement**
TRACe<x>:{FSETtling|PSETtling}:SELect	Selects a trace or queries the currently selected trace.
TRACe<x>:{FSETtling|PSETtling}:SHOW	Sets or queries whether to show or hide the trace.
TRACe:{FSETtling|PSETtling}:AVERage:ENABle	Sets or queries the number of data points for smoothing the trace.
TRACe<x>:PNOise:SELect	Selects the trace in the phase noise measurement.
TRACe<x>:PNOise:SHOW	Shows or hides the specified trace in the phase noise view.
TRACe<x>:PNOise:SMOothing:COUNt	Sets or queries the number of data points to take the moving average for smoothing the trace.
TRACe<x>:PNOise:SMOothing:ENABle	Sets or queries whether to enable smoothing the specified trace.
TRACe<x>:PNOise:SMOothing:RESet	Restarts the smoothing process.
TRACe:{FSETtling|PSETtling} subgroup (Option 12 only)	**Settling time measurements**
TRACe:{FSETtling|PSETtling}:AVERage:COUNt	Sets or queries the number of traces to combine.
TRACe:{FSETtling|PSETtling}:AVERage:COUNt:CURRent?	Queries the current running average count value.
TRACe:{FSETtling|PSETtling}:AVERage:ENABle	Determines whether to enable or disable averaging the trace(s).
TRACe:{FSETtling|PSETtling}:RESet	If trace averaging is enabled, this command resets the current averaged trace to 0.
TRACe:{FSETtling|PSETtling}:SMOothing:COUNt	Sets or queries the number of data points to take the moving average for smoothing the traces.
TRACe:{FSETtling|PSETtling}:SMOothing:ENABle	Determines whether to enable or disable smoothing the trace(s).
TRACe<x>:{FSETtling|PSETtling}:FREeze	Determines whether to freeze the specified trace in the Settling Time measurement display.
TRACe<x>:{FSETtling|PSETtling}:SELect	Selects the trace in the settling time measurement.
TRACe<x>:{FSETtling|PSETtling}:SHOW	Determines whether to show or hide the specified trace in the settling time view.
TRACe:SEM subgroup	**Spectral Emissions Mask measurements**
TRACe:SEM:COUNt	Sets or queries how many acquisitions run in the single acquisition mode for multi-trace functions.
TRACe:SEM:COUNt:RESet	Clears the multi-function data and counter and restarts the process.
TRACe:SEM:FUNCtion	Sets or queries the trace function.
TRACe:SGRam subgroup	**Spectrogram measurement**
TRACe:SGRam:DETection	Sets or queries the method to be used for decimating traces to fit the on screen.
TRACe:SGRam:FREeze	Sets or queries whether or not to freeze the spectrogram display.
TRACe:SGRam:FUNCtion	Sets or queries the trace function for the specified trace in the spectrogram.

Table 2-19: Trace commands (cont.)

Header	Description
TRACe:SGRam:FUNCtion:TIME	Sets or queries the number of traces to combine for the trace function.
TRACe:SGRam:SELect:LINE	Sets or queries the line number to send to the spectrum display.
TRACe<x>:SPECtrum subgroup	**Spectrum measurement**
TRACe<x>:SPECtrum	Sets or queries whether to show or hide the specified trace.
TRACe<x>:SPECtrum:AVERage:COUNt	Sets or queries the number of traces to combine for averaging.
TRACe<x>:SPECtrum:AVERage:RESet	Clears the average data and resets the average counter.
TRACe<x>:SPECtrum:COUNt	Sets or queries the count for the Max or Min Hold trace.
TRACe<x>:SPECtrum:COUNt:ENABle	Sets or queries whether or not to enable the count for Max/Min Hold.
TRACe<x>:SPECtrum:COUNt:RESet	Clears the Max or Min Hold data and counter, and restarts the process.
TRACe<x>:SPECtrum:DETection	Sets or queries the display detector.
TRACe<x>:SPECtrum:FREeze	Sets or queries whether or not to freeze the display of the trace.
TRACe<x>:SPECtrum:FUNCtion	Sets or queries the trace function.
TRACe<x>:SPECtrum:LEFToperand	Sets or queries the left operand for the math trace.
TRACe<x>:SPECtrum:RIGHtoperand	Sets or queries the right operand for the math trace.
TRACe<x>:SPECtrum:SELect	Sets or queries the trace number to display the readout.
TRACe:SPURious subgroup	**Spurious measurement**
TRACe:SPURious:COUNt	Sets or queries the count for the Max Hold or Average trace.
TRACe:SPURious:COUNt:ENABle	Sets or queries whether to enable the count for Max Hold or Average.
TRACe:SPURious:COUNt:RESet	Clears Max Hold or Average data and counter, and restarts the process.
TRACe:SPURious:FREeze	Determines whether or not to freeze the display of the trace.
TRACe:SPURious:FUNCtion	Sets or queries the trace function.
TRACe:TOVerview subgroup	**All measurements**
TRACe1:TOVerview	Enables display of or queries the display status of the specified trace.
TRACe1:TOVerview:AVERage:COUNt	Sets or queries the number of traces averaged to generate the specified trace.
TRACe1:TOVerview:COUNt	Enables or queries the count set for the specified trace.
TRACe1:TOVerview:COUNt:ENABle	Enables or queries the Average count for the specified trace.
TRACe1:TOVerview:COUNt:RESet	Sets or queries whether or not to freeze the display of the trace.
TRACe1:TOVerview:DETection	Enables or queries the type of detection for the specified trace.
TRACe1:TOVerview:FREeze	Enables or queries a halt to acquisition updates for the specified trace.
TRACe1:TOVerview:FUNCTion	Enables or queries the selected Function for the specified trace.
TRACe:WLAN group	**Wireless LAN measurements**
TRACe:WLAN:FLATness:FREeze	Sets or queries whether to freeze the average trace on the WLAN Spectral Flatness display.
TRACe:WLAN:FLATness:SHOW	Displays or hides the trace on the WLAN Spectral Flatness display.
TRACe<x>:WLAN:CONSte	Turns the selected trace on or off in the Constellation plot display.
TRACe<x>:WLAN:CONSte:FREeze	Turns the freeze state of the selected trace on or off.

Table 2-19: Trace commands (cont.)

Header	Description
TRACe<x>:WLAN:CRESponse:SELect	Sets or queries the active trace in the WLAN Channel Response.
TRACe<x>:WLAN:EVM:FREeze	Turns the freeze state of the selected trace on or off.
TRACe<x>:WLAN:EVM[:SHOW]	Displays or hides the selected trace.
TRACe<x>:WLAN:MERRor:FREeze	Turns the freeze state of the selected trace on or off.
TRACe<x>:WLAN:MERRor[:SHOW]	Displays or hides the selected trace.
TRACe<x>:WLAN:PERRor:FREeze	Turns the freeze state of the selected trace on or off.
TRACe<x>:WLAN:PERRor[:SHOW]	Displays or hides the selected trace.
TRACe:LTE subgroup	
TRACe<x>:LTE:ACLR:COUNt	Sets or queries how many acquisitions run in the single acquisition mode for multitrace functions.
TRACe<x>:LTE:ACLR:COUNt:RESet	Clears the multifunction (Max Hold or Average) data and counter and restarts the process.
TRACe<x>:LTE:ACLR:FUNCtion	Sets or queries the trace function.
TRACe<x>:LTE:CONSte:SYNChronization:SEQUence:STATe?	Returns whether or not a measurement signal (Primary Synchronization Signal/Secondary Synchronization Signal) is selected for the specified trace.

Trace Mnemonics

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

Table 2-20: Trace mnemonics

Measurement display	TRACe1	TRACe2	TRACe3	TRACe4	TRACe5
Amplitude versus Time	Trace 1	Trace 2	Math trace	NA	NA
DPX spectrum	+Peak trace	-Peak trace	Average trace	Math trace	Bitmap trace
Phase noise	Trace 1	Trace 2	NA	NA	NA
Spectrum	Trace 1	Trace 2	Trace 3	Math trace	Spectrogram

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

Retrieving Response Message

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

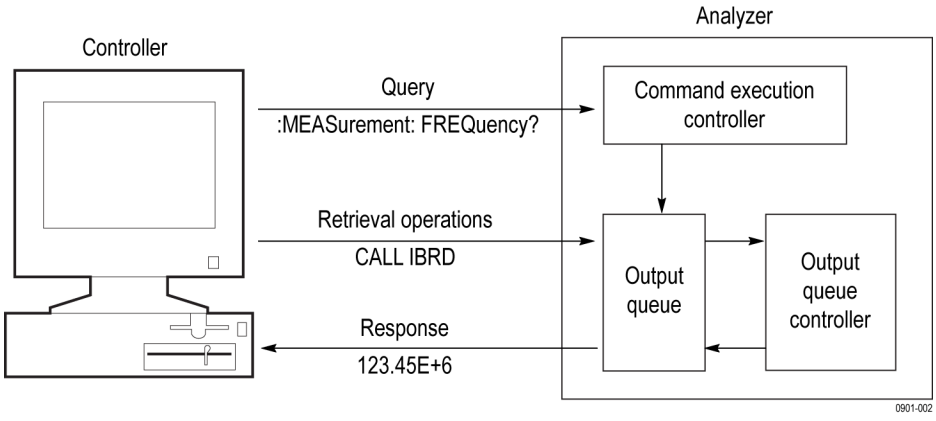


Figure 2-5: Retrieving response message

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

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

Command Descriptions

ABORt (No Query Form)

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

To start data acquisition, use the INITiate commands.

Conditions Measurement views: All

Group Abort commands

Syntax `ABORt`

Related Commands INITiate:CONTinuous, INITiate:RESume

Arguments None

Examples `ABORT` resets the trigger system and stops data acquisition.

*CAL (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

NOTE.

Conditions Measurement views: All

Group IEEE common commands

Syntax `*CAL`

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

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

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

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Channel power and ACPR

Group Calculate commands

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

Related Commands CALCulate:MARKer:ADD, CALCulate:ACPower:MARKer<x>:DELTa:Y?

Returns <NRf> Delta marker frequency for the selected marker.

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

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Channel power and ACPR

Group Calculate commands

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

Related Commands CALCulate:MARKer:ADD, CALCulate:ACPower:MARKer<x>:DELTa:X?

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

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

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

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

Conditions Measurement views: Channel power and ACPR

Group Calculate commands

Syntax CALCulate:ACPower:MARKer<x>:MAXimum

Arguments None

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

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

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

Conditions Measurement views: Channel power and ACPR

Group Calculate commands

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

Related Commands CALCulate:ACPower:MARKer<x>:PEAK:RIGHt

Arguments None

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

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

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

Conditions Measurement views: Channel power and ACPR

Group Calculate commands

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

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

Arguments None

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

CALCulate:ACPower:MARKer<x>:X

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

Conditions Measurement views: Channel power and ACPR

Group Calculate commands

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

Related Commands CALCulate:ACPower:MARKer<x>:Y?

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

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

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

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

Conditions Measurement views: Channel power and ACPR

Group Calculate commands

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

Related Commands CALCulate:ACPower:MARKer<x>:X

Arguments None

Returns <NRf> Marker amplitude of the selected marker.

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

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: General purpose analog demodulation

Group Calculate commands

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

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

Arguments None

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

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

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: General purpose analog demodulation

Group Calculate commands

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

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

Arguments None

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

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

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

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

Conditions Measurement views: General purpose analog demodulation

Group Calculate commands

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

Arguments None

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

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

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

Conditions Measurement views: General purpose analog demodulation

Group Calculate commands

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

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

Arguments None

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

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

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

Conditions Measurement views: General purpose analog demodulation

Group Calculate commands

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

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

Arguments None

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

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

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

Conditions Measurement views: General purpose analog demodulation

Group Calculate commands

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

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

Arguments None

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

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

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

Conditions Measurement views: General purpose analog demodulation

Group Calculate commands

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

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

Arguments None

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

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

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

Conditions Measurement views: General purpose analog demodulation

Group Calculate commands

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

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

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

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

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

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

Conditions Measurement views: General purpose analog demodulation

Group Calculate commands

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

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

Arguments None

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

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

CALCulate:AUDio:HARMonic:HNUMber

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

Conditions Measurement views: Audio Spectrum

Group Calculate commands

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

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

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

CALCulate:AUDio:HARMonic:NHNumber

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

Conditions Measurement views: Audio Spectrum

Group Calculate commands

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

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

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

CALCulate:AUDio:NHARmonic:EXCursion

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

Conditions Measurement views: Audio Spectrum

Group Calculate commands

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

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

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

CALCulate:AUDio:NHARmonic:IGNore

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

Conditions Measurement views: Audio Spectrum

Group Calculate commands

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

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

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

CALCulate:AUDio:NHARmonic:THReshold

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

Conditions Measurement views: Audio Spectrum

Group Calculate commands

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

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

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

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

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

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

Conditions Measurement views: Audio Spectrum

Group Calculate commands

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

Arguments None

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

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

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

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

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

Conditions Measurement views: Audio Spectrum

Group Calculate commands

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

Related Commands CALCulate:AUDio:SPECtrum:MARKer<x>:DELTa:X?

Arguments None

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

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

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

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

Conditions Measurement views: Audio Spectrum

Group Calculate commands

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

Arguments None

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

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

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

Conditions Measurement views: Audio Spectrum

Group Calculate commands

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

Related Commands CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:LOWer

Arguments None

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

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

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

Conditions Measurement views: Audio Spectrum

Group Calculate commands

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

Related Commands CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:RIGHt

Arguments None

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

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

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

Conditions Measurement views: Audio Spectrum

Group Calculate commands

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

Related Commands CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:HIGHer

Arguments None

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

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

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

Conditions Measurement views: Audio Spectrum

Group Calculate commands

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

Related Commands CALCulate:AUDio:SPECtrum:MARKer<x>:PEAK:LEFT

Arguments None

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

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

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

Conditions Measurement views: Audio Spectrum

Group Calculate commands

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

Arguments `TRACE1` places the specified marker on Trace 1.

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

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

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

Conditions Measurement views: Audio Spectrum

Group Calculate commands

Syntax

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

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

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

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

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

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

Conditions Measurement views: Audio Spectrum

Group Calculate commands

Syntax

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

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

Arguments None

Returns <NRf> the marker amplitude for the selected marker.

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

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Amplitude versus Time

Group Calculate commands

Syntax `CALCulate:AVTime:MARKer<x>:DELTa:X?`

Related Commands CALCulate:MARKer:ADD, CALCulate:AVTime:MARKer<x>:DELTa:Y?

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

Examples `CALCULATE:AVTIME:MARKER1:DELTA:X?` might return `38.0E-9`, indicating that the delta marker time is 38.0 ns.

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Amplitude versus Time

Group Calculate commands

Syntax `CALCulate:AVTime:MARKer<x>:DELTa:Y?`

Related Commands CALCulate:MARKer:ADD, CALCulate:AVTime:MARKer<x>:DELTa:X?

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

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

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

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

Conditions Measurement views: Amplitude versus Time

Group Calculate commands

Syntax `CALCulate:AVTime:MARKer<x>:MAXimum`

Arguments None

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

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

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

Conditions Measurement views: Amplitude versus Time

Group Calculate commands

Syntax `CALCulate:AVTime:MARKer<x>:PEAK:HIGHer`

Related Commands CALCulate:AVTime:MARKer<x>:PEAK:LOWer

Arguments None

Examples `CALCULATE:AVTIME:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

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

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

Conditions Measurement views: Amplitude versus Time

Group Calculate commands

Syntax `CALCulate:AVTime:MARKer<x>:PEAK:LEFT`

Related Commands CALCulate:AVTime:MARKer<x>:PEAK:RIGHt

Arguments None

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

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

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

Conditions Measurement views: Amplitude versus Time

Group Calculate commands

Syntax `CALCulate:AVTime:MARKer<x>:PEAK:LOWer`

Related Commands CALCulate:AVTime:MARKer<x>:PEAK:HIGHer

Arguments None

Examples `CALCULATE:AVTIME:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

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

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

Conditions Measurement views: Amplitude versus Time

Group Calculate commands

Syntax
```
CALCulate:AVTime:MARKer<x>:PEAK:RIGHt
```

Related Commands CALCulate:AVTime:MARKer<x>:PEAK:LEFT

Arguments None

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

CALCulate:AVTime:MARKer<x>:TRACe

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

Conditions Measurement views: Amplitude versus Time

Group Calculate commands

Syntax
```
CALCulate:AVTime:MARKer<x>:TRACe { TRACE1 | TRACE2 | TRACE3
| TRACE4 }
CALCulate:AVTime:MARKer<x>:TRACe?
```

Arguments

`TRACE1` places the selected marker on Trace 1.

`TRACE2` places the selected marker on Trace 2.

`TRACE3` places the selected marker on Trace 3.

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

Examples `CALCULATE:AVTIME:MARKER1:TRACE TRACE1` places Marker 1 (M1) on Trace 1.

CALCulate:AVTime:MARKer<x>:X

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Amplitude versus Time

Group Calculate commands

Syntax

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

Related Commands CALCulate:AVTime:MARKer<x>:Y?

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

Examples `CALCULATE:AVTIME:MARKER1:X 1.5U` places Marker 1 (M1) at 1.5 µs on the trace.

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Amplitude versus Time

Group Calculate commands

Syntax `CALCulate:AVTime:MARKer<x>:Y?`

Related Commands CALCulate:AVTime:MARKer<x>:X

Arguments None

Returns `<NRf>` Marker amplitude of the selected marker.

Examples `CALCULATE:AVTIME:MARKER1:Y?` might return `-2.73`, indicating Marker 1 (M1) is at -2.73 dBm.

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

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

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Bluetooth Constellation

Group Calculate commands

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

Arguments None

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

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

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

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

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

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Bluetooth Constellation.

Applicable when the Trace Type is set to Freq Dev.

Group Calculate commands

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

Related Commands The following commands are applicable when the Trace Type is set to IQ:
CALCulate:BLUETooth:CONSte:MARKer<x>:MAGNitude?
CALCulate:BLUETooth:CONSte:MARKer<x>:PHASe?
Use " " to change the Trace Type.

Arguments None

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

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

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

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

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Bluetooth Constellation

Applicable when the Trace Type is set to IQ.

Group Calculate commands

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

Arguments None

Returns <NRf> The magnitude for the selected marker.

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

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

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

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Bluetooth Constellation

Group Calculate commands

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

Arguments None

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

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

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

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Bluetooth Constellation

Group Calculate commands

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

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

Arguments None

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

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

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

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Bluetooth Constellation

Group Calculate commands

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

Related Commands CALCulate:BLUEtooth:CONSte:MARKer<x>:PEAK:LEFT

Arguments None

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

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

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

Conditions Measurement views: Bluetooth Constellation.

Applicable when the Trace Type is set to IQ.

Group Calculate commands

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

Arguments None

Returns <NRf> The phase readout for the selected marker in degrees between -180 and +180.

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

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

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

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Bluetooth Constellation

Group Calculate commands

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

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

Arguments None

Returns <NRf> The symbol readout for the selected marker.

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

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

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

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Bluetooth Constellation

Group Calculate commands

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

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

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

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

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

Conditions Measurement views: Bluetooth Constellation

Group Calculate commands

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

Arguments None

Returns <NRf> The value readout for the selected marker.

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

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

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

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Bluetooth Constellation

Group Calculate commands

Syntax

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

Arguments <value> ::= <NRf> specifies the time position of the marker.

Range (analysis offset) to [(analysis offset) + (analysis length)].

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

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

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

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Bluetooth eye diagram

Group Calculate commands

Syntax

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

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

Arguments None

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

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

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

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

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

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Bluetooth eye diagram

Group Calculate commands

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

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

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

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

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

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

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Bluetooth eye diagram

Group Calculate commands

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

Arguments None

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

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

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

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Bluetooth eye diagram

Group Calculate commands

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

Related Commands CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:LOWer

Arguments None

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

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

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

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Bluetooth eye diagram

Group Calculate commands

Syntax `CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:LEFT`

Related Commands CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:RIGHt

Arguments None

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

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

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

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Bluetooth eye diagram

Group Calculate commands

Syntax `CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:LOWer`

Related Commands CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:HIGHer

Arguments None

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

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

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

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Bluetooth eye diagram

Group Calculate commands

Syntax `CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:RIGHt`

Related Commands CALCulate:BLUEtooth:EDIagram:MARKer<x>:PEAK:LEFT

Arguments None

Examples `CALCulate:BLUEtooth:EDIagram:MARKer1:PEAK:RIGHT` moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:BLUEtooth:EDIagram:MARKer<x>:TRACe

The command places the selected marker on the I or Q trace in the Bluetooth eye diagram display. The query returns the name of the trace where the current marker resides.

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Bluetooth eye diagram

Group Calculate commands

Syntax
```
CALCulate:BLUEtooth:EDIagram:MARKer<x>:TRACe <Enum>
CALCulate:BLUEtooth:EDIagram:MARKer<x>:TRACe?
```

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

Trace1 specifies the I trace and Trace2 specifies the Q trace.

Returns For I and Q traces, TRACE1 identifies the I trace and TRACE2 identifies the Q trace.

Examples `CALCulate:BLUEtooth:EDIagram:MARKer1:TRACe TRACE1` places Marker 1 (M1) on the I trace.

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

Sets or queries the horizontal position (time) of the selected marker in the Bluetooth eye diagram measurement.

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Bluetooth eye diagram

Group Calculate commands

Syntax

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

Related Commands CALCulate:BLUEtooth:EDIagram:MARKer<x>:Y?

Arguments <value> ::= <NRf> the horizontal position (time) of the marker.

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

Examples `CALCulate:BLUEtooth:EDIagram:MARKer1:X:TIME 38.5` places Marker 1 (M1) at 38.5 symbols on the trace.

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

Returns the vertical position of the selected marker in the Bluetooth Eye diagram measurement.

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Bluetooth eye diagram

Group Calculate commands

Syntax

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

Related Commands CALCulate:BLUEtooth:EDIagram:MARKer<x>:X[:TIME]

Arguments None

Returns <value> ::= <NRf> the vertical position of the selected marker. The units are Hz if the Trace Type is set to Freq Dev; if the Trace Type is set to IQ, there are no units.

Examples `CALCulate:BLUEtooth:EDIagram:MARKer1:Y?` Might return 571.8E-3, indicating Marker 1 (M1) is at 0.5718.

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

Queries the delta marker time for the selected marker on the Frequency Deviation vs. Time display

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

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

Conditions Measurement views: Frequency Deviation vs. Time

Group Calculate commands

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

Related Commands CALCulate:BLUEtooth:FDVTIME:MARKer<x>:DELTa:Y?

Arguments None

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

Examples `CALCULATE:BLUETOOTH:FDVTIME:MARKER1:DELTa:X:TIME?` might return 62.75, indicating that the delta marker time is 62.75 symbols.

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

Returns the delta marker amplitude for the selected marker in the Frequency Deviation vs. Time display.

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Frequency Deviation vs. Time

Group Calculate commands

Syntax `CALCulate:BLUEtooth:FDVTIME:MARKer<x>:DELTa:Y?`

Related Commands CALCulate:BLUEtooth:FDVTIME:MARKer<x>:DELTa:X[:TIME]?

Arguments None

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

Examples `CALCulate:BLUEtooth:FDVTIME:MARKer1:DELTa:Y?` return -1.043, indicating that the delta marker amplitude is -1.043.

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

Moves the selected marker to the highest peak on the trace in the Frequency Deviation vs. Time display.

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

Conditions Measurement views: Frequency Deviation vs. Time

Group Calculate commands

Syntax `CALCulate:BLUEtooth:FDVTime:MARKer<x>:MAXimum`

Arguments None

Examples `CALCULATE:BLUETOOTH:FDVTIME:MARKER1:MAXIMUM` moves Marker 1 (M1) to the highest peak on the trace.

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

Moves the selected marker to the next peak higher in amplitude in the Frequency Deviation vs. Time display.

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

Conditions Measurement views: Frequency Deviation vs. Time

Group Calculate commands

Syntax `CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:HIGHer`

Related Commands CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:LOWer

Arguments None

Examples `CALCULATE:BLUETOOTH:FDVTIME:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

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

Moves the selected marker to the next peak to the left on the trace in the Frequency Deviation vs. Time display.

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

Conditions Measurement views: Frequency Deviation vs. Time

Group Calculate commands

Syntax `CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:LEFT`

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

Arguments None

Examples `CALCULATE:BLUETOOTH:FDVTIME:MARKER1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on Frequency Deviation vs. Time display.

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

Conditions Measurement views: Frequency Deviation vs. Time

Group Calculate commands

Syntax `CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:LOWer`

Related Commands CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:HIGHer

Arguments None

Examples `CALCULATE:BLUETOOTH:FDVTIME:MARKER1:PEAK:LOWER`moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:RIGHT (No Query Form)

Moves the selected marker to the next peak to the right on the trace in the Frequency Deviation vs. Time display.

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

Conditions Measurement views: Frequency Deviation vs. Time

Group Calculate commands

Syntax `CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:RIGHT`

Related Commands CALCulate:BLUEtooth:FDVTime:MARKer<x>:PEAK:LEFT

Arguments None

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

CALCulate:BLUEtooth:FDVTIME:MARKer<x>:X[:TIME]

Sets or queries the marker time in the Frequency Deviation vs. Time display.

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

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

Conditions Measurement views: Frequency Deviation vs. Time

Group Calculate commands

Syntax
```
CALCulate:BLUEtooth:FDVTIME:MARKer<x>:X[:TIME] <value>
CALCulate:BLUEtooth:FDVTIME:MARKer<x>:X[:TIME]?
```

Related Commands CALCulate:BLUEtooth:FDVTIME:MARKer<x>:Y

Arguments <value> ::= <NRf> specifies the marker time in seconds or in symbols based on the Units specified under the Analysis Time.

Examples `CALCULATE:BLUETOOTH:FDVTIME:MARKER1:X:TIME -234.5E-6` places Marker 1 (M1) at -234.5 µs on the trace.

CALCulate:BLUEtooth:FDVTIME:MARKer<x>:Y

Sets or queries the marker amplitude in the Frequency Deviation vs. Time display.

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Frequency Deviation vs. Time

Group Calculate commands

Syntax `CALCulate:BLUEtooth:FDVTIME:MARKer<x>:Y`

Related Commands CALCulate:BLUEtooth:FDVTIME:MARKer<x>:X[:TIME]

Arguments <value> ::= <NRf> specifies the marker frequency value in Hz.

Examples CALCULATE:BLUETOOTH:FDVTIME:MARKER1:Y? might return 571.8E-3 indicating Marker 1 (M1) is 0.5718 Hz.

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

Returns the delta marker frequency for the selected marker on the Bluetooth 20 dB bandwidth measurement.

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: BT 20 dB Bandwidth

Group Calculate commands

Syntax `CALCulate:BOBW:MARKer<x>:DELTa:X?`

Related Commands CALCulate:BOBW:MARKer<x>:DELTa:Y?

Arguments None

Returns <NRf>, Delta marker frequency for the selected marker in Hz.

Examples `CALCULATE:BOBW:MARKER1:DELTA:X?` might return 62.75E+3, indicating that the delta marker time is 62.75 kHz.

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

Returns the delta marker amplitude for the selected marker on the Bluetooth 20 dB bandwidth measurement.

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: BT 20 dB Bandwidth

Group Calculate commands

Syntax `CALCulate:BOBW:MARKer<x>:DELTa:Y?`

Related Commands CALCulate:BOBW:MARKer<x>:DELTa:X?

Arguments None

Returns <NRf>, the Delta marker amplitude for the selected marker.

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

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

Positions the selected marker at the maximum value of the display.

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: BT 20 dB Bandwidth

Group Calculate commands

Syntax `CALCulate:BOBW:MARKer<x>:MAXimum`

Arguments None

Examples CALCULATE:BOBW:MARKER1:MAXIMUM positions Marker 1 (M1) at the maximum value of the display.

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

Moves the selected marker to the next peak higher in amplitude on the BT 20dB bandwidth trace.

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: BT 20 dB Bandwidth

Group Calculate commands

Syntax `CALCulate:BOBW:MARKer<x>:PEAK:HIGHer`

Related Commands CALCulate:BOBW:MARKer<x>:PEAK:LOWer

Arguments None

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

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

Moves the selected marker in the time domain to the left, relative to the previous marker position on the trace.

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: BT 20 dB Bandwidth

Group Calculate commands

Syntax `CALCulate:BOBW:MARKer<x>:PEAK:LEFT`

Arguments None

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

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

Moves the selected marker to the next peak lower in amplitude on the BT 20dB bandwidth trace.

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: BT 20 dB Bandwidth

Group Calculate commands

Syntax `CALCulate:BOBW:MARKer<x>:PEAK:LOWer`

Arguments None

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

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

Moves the selected marker in the time domain to the right, relative to the previous marker position on the trace.

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: BT 20 dB Bandwidth

Group Calculate commands

Syntax `CALCulate:BOBW:MARKer<x>:PEAK:RIGHt`

Arguments None

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

CALCulate:BOBW:MARKer<x>[:SET]:CENTer (No Query Form)

Sets the center frequency to the value at the marker position in the BT 20 dB Bandwidth view.

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: BT 20 dB Bandwidth

Group Calculate commands

Syntax

```
CALCulate:BOBW:MARKer<x>[:SET]:CENTer
```

Arguments None

Examples CALCULATE:BOBW:MARKER1:SET:CENTER sets the center frequency to the value at the Marker 1 (M1) position.

CALCulate:BOBW:MARKer<x>:X

Sets or queries the time position of the selected marker on the BT 20 dB Bandwidth measurement.

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: BT 20 dB Bandwidth

Group Calculate commands

Syntax

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

Related Commands CALCulate:BOBW:MARKer<x>:Y?

Arguments <value> ::= <NRf> specifies the horizontal position of the marker.

Range: Start to Stop frequency (left to right edge of the horizontal axis). Using an out-of-range value causes an execution error (-222, "Data out of range").

Examples CALCULATE:BOBW:MARKER1:X 2.4800000000E+9 places the Marker 1 (M1) at 2.4800000000E+9 on the BT 20dB Bandwidth trace.

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

Returns the amplitude position of the selected marker on the Bluetooth 20 dB Bandwidth trace.

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: BT 20 dB Bandwidth

Group Calculate commands

Syntax `CALCulate:BOBW:MARKer<x>:Y?`

Related Commands CALCulate:BOBW:MARKer<x>:DELTa:X?

Arguments None

Returns <value> ::= <NRf> specifies the vertical position of the marker.

Examples CALCULATE:BOBW:MARKER1:Y? returns the marker amplitude in dB on the BT 20dB Bandwidth trace.

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

Returns the delta marker time for the selected marker on the Constellation trace.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Constellation

Group Calculate commands

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

Returns `<NRf>` Delta marker time for the selected marker.
The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples `CALCULATE:CONSTE:MARKER1:DELTA:X:TIME?` might return `-62.75`, indicating that the delta marker time is -62.75 symbols.

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

Queries the frequency deviation of the selected marker in the Constellation measurement for an FSK modulated signal.

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

Conditions The command is valid for 2, 4, 8, 16 FSK or C4FM modulated signals.

Group Calculate commands

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

Related Commands CALCulate:CONSte:MARKer<x>:MAGNitude?

CALCulate:CONSte:MARKer<x>:PHASe?

Arguments None

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

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

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Constellation

Group Calculate commands

Syntax `CALCulate:CONSte:MARKer<x>:MAGNitude?`

Related Commands CALCulate:CONSte:MARKer<x>:X

Arguments None

Returns `<NRf>` The magnitude readout for the selected marker.

Examples `CALCulate:CONSte:MARKer1:MAGNitude?` might return `0.713927`, indicating the magnitude readout of Marker 1 (M1) is 0.713927.

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Constellation

Group Calculate commands

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

Arguments None

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

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Constellation

Group Calculate commands

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

Related Commands CALCulate:CONSte:MARKer<x>:PEAK:RIGHt

Arguments None

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

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Constellation

Group Calculate commands

Syntax `CALCulate:CONSte:MARKer<x>:PEAK:RIGHt`

Related Commands CALCulate:CONSte:MARKer<x>:PEAK:LEFT

Arguments None

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

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Constellation

Group Calculate commands

Syntax `CALCulate:CONSte:MARKer<x>:PHASe?`

Related Commands CALCulate:CONSte:MARKer<x>:X

Arguments None

Returns `<NRf>` The phase readout for the selected marker.

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

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

Queries the symbol readout of the selected marker in the Constellation measurement.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Constellation

Group Calculate commands

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

Related Commands CALCulate:CONSte:MARKer<x>:X

Arguments None

Returns `<NRf>` The symbol readout for the selected marker.

Examples `CALCULATE:CONSTE:MARKER1:SYMBOL?` might return `62.00`, indicating the symbol readout of Marker 1 (M1) is 62.

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Constellation

Group Calculate commands

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

Related Commands CALCulate:CONSte:MARKer<x>:X

Arguments None

Returns `<NRf>` The value readout for the selected marker.

Examples `CALCULATE:CONSTE:MARKER1:VALUE?` might return `2.00`, indicating the value readout of Marker 1 (M1) is 2.

CALCulate:CONSte:MARKer<x>:X

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Constellation

Group Calculate commands

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

Related Commands CALCulate:CONSte:MARKer<x>:MAGNitude?, CALCulate:CONSte:MARKer<x>:PHASe?, CALCulate:CONSte:MARKer<x>:SYMBol?

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

Examples `CALCULATE:CONSTE:MARKER1:X -1.63875m` places the Marker 1 (M1) at -1.63875 ms on the Constellation trace.

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

Returns the delta marker time for the selected marker on the Demodulated I&Q versus Time trace.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Demodulated I&Q versus Time

Group Calculate commands

Syntax `CALCulate:DIQVtime:MARKer<x>:DELTa:X[:TIME]?`

Related Commands CALCulate:DIQVtime:MARKer<x>:DELTa:Y?

Arguments None

Returns <NRf> Delta marker time for the selected marker.
Use the [SENSe]:DDEMod:TIMe:UNITs command to select the time unit: symbols (default) or seconds.

Examples `CALCULATE:DIQVTIME:MARKER1:DELTA:X:TIME?` might return `62.75`, indicating that the delta marker time is 62.75 symbols.

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

Returns the delta marker amplitude for the selected marker on the Demodulated I&Q versus Time trace.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Demodulated I&Q versus Time

Group Calculate commands

Syntax `CALCulate:DIQVtime:MARKer<x>:DELTa:Y?`

Related Commands CALCulate:DIQVtime:MARKer<x>:DELTa:X[:TIME]?

Arguments None

Returns <NRf> Delta marker amplitude for the selected marker in volts.

Examples `CALCULATE:DIQVTIME:MARKER1:DELTA:Y?` might return `-78.24E-3`, indicating that the delta marker amplitude is -78.24 mV.

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

Moves the selected marker to the highest peak on the Demodulated I&Q versus Time trace.

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

Conditions Measurement views: Demodulated I&Q versus Time

Group Calculate commands

Syntax `CALCulate:DIQVtime:MARKer<x>:MAXimum`

Arguments None

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

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

Moves the selected marker to the next peak higher in amplitude on the Demodulated I&Q versus Time trace.

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

Conditions Measurement views: Demodulated I&Q versus Time

Group Calculate commands

Syntax `CALCulate:DIQVtime:MARKer<x>:PEAK:HIGHer`

Related Commands CALCulate:DIQVtime:MARKer<x>:PEAK:LOWer

Arguments None

Examples `CALCULATE:DIQVTIME:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

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

Moves the selected marker to the next peak to the left on the Demodulated I&Q versus Time trace.

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

Conditions Measurement views: Demodulated I&Q versus Time

Group Calculate commands

Syntax `CALCulate:DIQVtime:MARKer<x>:PEAK:LEFT`

Related Commands CALCulate:DIQVtime:MARKer<x>:PEAK:RIGHt

Arguments None

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

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

Moves the selected marker to the next peak lower in amplitude on the Demodulated I&Q versus Time trace.

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

Conditions Measurement views: Demodulated I&Q versus Time

Group Calculate commands

Syntax `CALCulate:DIQVtime:MARKer<x>:PEAK:LOWer`

Related Commands CALCulate:DIQVtime:MARKer<x>:PEAK:HIGHer

Arguments None

Examples `CALCULATE:DIQVTIME:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

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

Moves the selected marker to the next peak to the right on the Demodulated I&Q versus Time trace.

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

Conditions Measurement views: Demodulated I&Q versus Time

Group Calculate commands

Syntax

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

Related Commands CALCulate:DIQVtime:MARKer<x>:PEAK:LEFT

Arguments None

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

CALCulate:DIQVtime:MARKer<x>:TRACe

Places the selected marker on the Demodulated I&Q versus Time trace. The query returns the name of the trace on which the marker resides.

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

Conditions Measurement views: Demodulated I&Q versus Time

Group Calculate commands

Syntax

```
CALCulate:DIQVtime:MARKer<x>:TRACe <Enum>
CALCulate:DIQVtime:MARKer<x>:TRACe?
```

Related Commands CALCulate:IQVTime:MARKer<x>:TRACe

Arguments `<Enum>::= TRACE1 | TRACE2` specifies which trace to place the marker on. TRACE1 specifies the "I" trace and TRACE2 specifies the "Q" trace.

Returns For I and Q traces, TRACE1 identifies the "I" trace and TRACE2 identifies the "Q" trace.

Examples `CALCulate:DIQVtime:MARKer1:TRACe TRACE1` places Marker 1 (M1) on the I trace.

CALCulate:DIQVtime:MARKer<x>:X[:TIME]

Sets or queries the horizontal position (time) of the selected marker in the Demodulated I&Q versus Time measurement.

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

Conditions Measurement views: Demodulated I&Q versus Time

Group Calculate commands

Syntax
```
CALCulate:DIQVtime:MARKer<x>:X[:TIME] <value>
CALCulate:DIQVtime:MARKer<x>:X[:TIME]?
```

Related Commands CALCulate:DIQVtime:MARKer<x>:Y?

Arguments `<value>::=<NRf>` specifies the horizontal position (time) of the marker. Use the [SENSe]:DDEMod:TIMe:UNITs command to select the time unit: symbols (default) or seconds.

Examples `CALCULATE:DIQVTIME:MARKER1:X:TIME 38.5` places Marker 1 (M1) at 38.5 symbols on the trace.

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

Queries the marker amplitude of the selected marker in the Demod I&Q versus Time measurement.

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

Conditions Measurement views: Demodulated I&Q versus Time

Group Calculate commands

Syntax `CALCulate:DIQVtime:MARKer<x>:Y?`

Related Commands CALCulate:DIQVtime:MARKer<x>:X[:TIME]

Arguments None

Returns `<NRf>` Marker amplitude of the selected marker in volts.

Examples `CALCulate:DIQVtime:MARKer1:Y?` might return `25.803E-3`, indicating Marker 1 (M1) is at 25.803 mV.

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command and attached to the Bitmap trace using the CALCulate:DPX:MARKer<x>:TRACe command.

Conditions Measurement views: DPX spectrum

Group Calculate commands

Syntax `CALCulate:DPX:MARKer<x>:DELTa:X:AMPLitude?`

Related Commands CALCulate:DPX:MARKer<x>:X[:FREQuency], CALCulate:DPX:MARKer<x>:Y?

Arguments None

Returns <NRf> Delta marker amplitude for the selected marker in dB.

Examples CALCULATE:DPX:MARKER1:DELTA:X:AMPLITUDE? might return 35.12, indicating that the delta marker amplitude is 35.12 dB for Marker 1 (M1).

CALCulate:DPX:MARKer<x>:DELTa:X[:FREQuency]? (Query Only)

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: DPX spectrum

Group Calculate commands

Syntax CALCulate:DPX:MARKer<x>:DELTa:X[:FREQuency]?

Related Commands CALCulate:DPX:MARKer<x>:DELTa:X:AMPLitude?, CALCulate:DPX:MARKer<x>:DELTa:Y?

Arguments None

Returns <NRf> Delta marker frequency for the selected marker in Hz.

Examples CALCULATE:DPX:MARKER1:DELTA:X? might return 1.28E+6, indicating that the delta marker frequency is 1.28 MHz for Marker 1 (M1).

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: DPX spectrum

Group Calculate commands

Syntax `CALCulate:DPX:MARKer<x>:DELTa:Y?`

Related Commands CALCulate:DPX:MARKer<x>:DELTa:X[:FREQuency]?

Arguments None

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

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

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: DPX spectrum

Group Calculate commands

Syntax `CALCulate:DPX:MARKer<x>:MAXimum`

Arguments None

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

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: DPX spectrum

Group Calculate commands

Syntax `CALCulate:DPX:MARKer<x>:PEAK:HIGHer`

Related Commands CALCulate:DPX:MARKer<x>:PEAK:LOWer

Arguments None

Examples `CALCULATE:DPX:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: DPX spectrum

Group Calculate commands

Syntax `CALCulate:DPX:MARKer<x>:PEAK:LEFT`

Related Commands CALCulate:DPX:MARKer<x>:PEAK:RIGHt

Arguments None

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

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: DPX spectrum

Group Calculate commands

Syntax `CALCulate:DPX:MARKer<x>:PEAK:LOWer`

Related Commands CALCulate:DPX:MARKer<x>:PEAK:HIGHer

Arguments None

Examples `CALCULATE:DPX:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: DPX spectrum

Group Calculate commands

Syntax `CALCulate:DPX:MARKer<x>:PEAK:RIGHt`

Related Commands CALCulate:DPX:MARKer<x>:PEAK:LEFT

Arguments None

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

CALCulate:DPX:MARKer<x>:POWer:DENSity? (Query Only)

Returns the power density of the specified marker in the DPX measurement.

Conditions Measurement view: DPX

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid.

Group Calculate commands

Syntax `CALCulate:DPX:MARKer<x>:POWer:DENSity?`

Related Commands

Returns <NRf> is the power density of the selected marker in dBm/Hz.

Examples `CALCULATE:DPX:MARKER1:POWER:DENSITY?` might return 80.2432546111, indicating the power density is 80.24 dBm/Hz.

CALCulate:DPX:MARKer<x>:POWer:IDensity? (Query Only)

Returns the integrated power density reading between the MR (reference marker) and the selected marker in the DPX measurement.

Conditions Measurement view: DPX

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid.

Group Calculate commands

Syntax `CALCulate:DPX:MARKer<x>:POWer:IDensity?`

Returns <NRf> is the integrated power density reading between the MR (reference marker) and the selected marker in dBm.

Examples `CALCULATE:DPSA:MARKER1:POWER:REFDENSITY?` might return -78.56 dBm/Hz.

CALCulate:DPX:MARKer<x>:POWer:INTEgrated? (Query Only)

Returns the integrated power reading between the MR (reference marker) and the selected marker in the DPX measurement.

Conditions Measurement view: DPX

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid.

Group Calculate commands

Syntax `CALCulate:DPX:MARKer<x>:POWer:INTEgrated?`

Returns <NRf> is the integrated power reading between MR (reference marker) and the selected marker in dBm.

Examples `CALCULATE:DPX:MARKER1:POWER:INTEGRATED?` might return -14.66, indicating the integrated power is -14.66 dBm.

CALCulate:DPX:MARKer<x>[:SET]:CENTer (No Query Form)

Sets the center frequency to the marker frequency in the DPX spectrum view.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: DPX spectrum

Group Calculate commands

Syntax `CALCulate:DPX:MARKer<x>[:SET]:CENTer`

Arguments None

Examples `CALCULATE:DPX:MARKER1:SET:CENTER` sets the center frequency to the marker frequency in the DPX spectrum view.

CALCulate:DPX:MARKer<x>:TRACe

Sets or queries the trace to attach the specified marker to in the DPX spectrum measurement.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: DPX spectrum

Group Calculate commands

Syntax

```
CALCulate:DPX:MARKer<x>:TRACe { BITMAP | TRACE1 | TRACE2 |
TRACE3 | TRACE4 }
CALCulate:DPX:MARKer<x>:TRACe?
```

Arguments `BITMAP` attaches the specified marker to the Bitmap trace.

`TRACE1` attaches the specified marker to the +Peak trace.

`TRACE2` attaches the specified marker to the -Peak trace.

`TRACE3` attaches the specified marker to the Average trace.

`TRACE4` attaches the specified marker to the Math trace.

Examples `CALCULATE:DPX:MARKER1:TRACE TRACE1` attaches Marker 1 (M1) to the +Peak trace.

CALCulate:DPX:MARKer<x>:X:AMPLitude

Sets or queries the amplitude position of the selected marker in the DPX spectrum view. This command is valid for the marker on the bitmap trace (refer to the CALCulate:DPX:MARKer<x>:TRACe command). The frequency position is set by the CALCulate:DPX:MARKer<x>:X[:FREQuency] command.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: DPX spectrum

Group Calculate commands

Syntax
```
CALCulate:DPX:MARKer<x>:X:AMPLitude <value>
CALCulate:DPX:MARKer<x>:X:AMPLitude?
```

Arguments <value>::=<NRf> specifies the amplitude position of the marker.
Range: -100 to 0 dBm.

Examples `CALCULATE:DPX:MARKER1:X:AMPLITUDE –34.5dBm` places Marker 1 (M1) at –34.5 dBm.

CALCulate:DPX:MARKer<x>:X[:FREQuency]

Sets or queries the frequency position of the selected marker in the DPX spectrum view.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

The command is only valid for the Spectrum and Frequency plots.

Conditions Measurement views: DPX spectrum

Group Calculate commands

Syntax
```
CALCulate:DPX:MARKer<x>:X[:FREQuency] <value>
CALCulate:DPX:MARKer<x>:X[:FREQuency]?
```

Related Commands CALCulate:DPX:MARKer<x>:Y?

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

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

CALCulate:DPX:MARKer<x>:X:PHASe

Sets or queries the phase position of the selected marker in the DPX phase plot.

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

Conditions Measurement views: DPX spectrum

Group Calculate commands

Syntax

```
CALCulate:DPX:MARKer<x>:X:PHASe <value>
CALCulate:DPX:MARKer<x>:X:PHASe?
```

Arguments <value>::<NRf> phase setting in degrees for the specified marker.

Examples `CALCulate:DPX:MARKer0:X:PHASe?` might return `94.0000000` indicating the phase for the reference marker (M0) is 94.00°.

CALCulate:DPX:MARKer<x>:X:TIME

Sets or queries the time position of the selected marker in the DPX spectrum view. The parameter <x> = 0 to 4. The specified marker must be activated using the CALCulate:MARKer:ADD command. The command is valid for the Amplitude, Frequency, and Phase plots.

Conditions Measurement views: DPX spectrum

Group Calculate commands

Syntax

```
CALCulate:DPX:MARKer<x>:X:TIME <value>
CALCulate:DPX:MARKer<x>:X:TIME?
```

Arguments <value>::<NRf> Time position of the specified marker.

Examples `CALCulate:DPX:MARKer0:X:TIME?` might return `1.000000000E-3` indicating the time position of the reference marker (MR0) is 1.000 ms.

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

Queries the vertical position of the selected marker in the DPX spectrum view. The data occurrence rate is returned for the bitmap trace, and the amplitude value for the +peak, -peak, average, and math traces. The horizontal position can be set by the CALCulate:DPX:MARKer<x>:X:AMPLitude and CALCulate:DPX:MARKer<x>:X[:FREQuency] commands.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: DPX spectrum

Group Calculate commands

Syntax `CALCulate:DPX:MARKer<x>:Y?`

Arguments None

Returns <NRf>The value type depends on which trace the marker is placed on (refer to the CALCulate:DPX:MARKer<x>:TRACe command):

The amplitude value is returned in dBm for the marker on the +peak, -peak, average, or math trace. The data occurence rate is returned in percent (%) for the marker on the bitmap trace.

Examples `CALCULATE:DPX:MARKER1:Y?` might return `-34.28` indicating Marker 1 (M1) is at -34.28 dBm when it is placed on the +peak, -peak, average, or math trace.

`CALCULATE:DPX:MARKER1:Y?` might return `76.5` indicating Marker 1 (M1) is at 76.5% when it is placed on the bitmap trace.

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

Returns the delta marker time for the selected marker on the eye diagram trace.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Eye diagram

Group Calculate commands

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

Related Commands CALCulate:EDIagram:MARKer<x>:DELTa:Y?

Arguments None

Returns <NRf> Delta marker time for the selected marker.
Use the [SENSe]:DDEMod:TIMe:UNITs command to select the time unit: symbols (default) or seconds.

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

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

Returns the delta marker amplitude for the selected marker on the eye diagram trace.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Eye diagram

Group Calculate commands

Syntax `CALCulate:EDIagram:MARKer<x>:DELTa:Y?`

Related Commands CALCulate:EDIagram:MARKer<x>:DELTa:X[:TIME]?

Arguments None

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

Examples `CALCULATE:EDIAGRAM:MARKER1:DELTA:Y?` might return `-1.043`, indicating that the delta marker amplitude is -1.043.

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

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

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

Conditions Measurement views: Eye diagram

Group Calculate commands

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

Arguments None

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

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

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

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

Conditions Measurement views: Eye diagram

Group Calculate commands

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

Related Commands CALCulate:EDIagram:MARKer<x>:PEAK:LOWer

Arguments None

Examples `CALCULATE:EDIAGRAM:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

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

Moves the selected marker to the next peak to the left on the eye diagram trace.

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

Conditions Measurement views: Eye diagram

Group Calculate commands

Syntax `CALCulate:EDIagram:MARKer<x>:PEAK:LEFT`

Related Commands CALCulate:EDIagram:MARKer<x>:PEAK:RIGHt

Arguments None

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

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

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

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

Conditions Measurement views: Eye diagram

Group Calculate commands

Syntax `CALCulate:EDIagram:MARKer<x>:PEAK:LOWer`

Related Commands CALCulate:EDIagram:MARKer<x>:PEAK:HIGHer

Arguments None

Examples `CALCULATE:EDIAGRAM:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

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

Moves the selected marker to the next peak to the right on the eye diagram trace.

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

Conditions Measurement views: Eye diagram

Group Calculate commands

Syntax `CALCulate:EDIagram:MARKer<x>:PEAK:RIGHt`

Related Commands CALCulate:EDIagram:MARKer<x>:PEAK:LEFT

Arguments None

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

CALCulate:EDIagram:MARKer<x>:TRACe

Places the selected marker on the I or Q trace in the eye diagram display. The query returns the name of the trace on which the marker resides. Valid on all modulation types except 2|4|8|16FSK or C4FM.

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

Conditions Measurement views: Eye diagram

Group Calculate commands

Syntax
```
CALCulate:EDIagram:MARKer<x>:TRACe <Enum>
CALCulate:EDIagram:MARKer<x>:TRACe?
```

Related Commands CALCulate:EDIagram:MARKer<x>:X[:TIME]

Arguments `<Enum>::= TRACE1 | TRACE2` specifies which trace to place the marker on. TRACE1 specifies the "I" trace and TRACE2 specifies the "Q" trace.

Returns For I and Q traces, TRACE1 identifies the "I" trace and TRACE2 identifies the "Q" trace.

Examples `CALCulate:EDIagram:MARKer1:TRACe TRACE1` places Marker 1 (M1) on the I trace in an eye diagram.

CALCulate:EDIagram:MARKer<x>:X[:TIME]

Sets or queries the horizontal position (time) of the selected marker in the eye diagram measurement.

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

Conditions Measurement views: Eye diagram

Group Calculate commands

Syntax
```
CALCulate:EDIagram:MARKer<x>:X[:TIME] <value>
CALCulate:EDIagram:MARKer<x>:X[:TIME]?
```

Related Commands CALCulate:EDIagram:MARKer<x>:Y?

Arguments `<value>::=<NRf>` specifies the horizontal position (time) of the marker. Use the [SENSe]:DDEMod:TIMe:UNITs command to select the time unit: symbols (default) or seconds.

Examples `CALCulate:EDIagram:MARKer1:X:TIME 38.5` places Marker 1 (M1) at 38.5 symbols on the trace.

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

Queries the vertical position of the selected marker in the eye diagram measurement.

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

Conditions Measurement views: Eye diagram

Group Calculate commands

Syntax `CALCulate:EDIagram:MARKer<x>:Y?`

Related Commands CALCulate:EDIagram:MARKer<x>:X[:TIME]

Arguments None

Returns `<NRf>` Vertical position of the selected marker.

Examples `CALCULATE:EDIAGRAM:MARKER1:Y?` might return `571.8E-3`, indicating Marker 1 (M1) is at 0.5718.

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: EVM versus Time

Group Calculate commands

Syntax `CALCulate:EVM:MARKer<x>:DELTa:X?`

Related Commands CALCulate:EVM:MARKer<x>:DELTa:Y?

Arguments None

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

The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples CALCulate:EVM:MARKer1:DELTa:X? might return 9.52, indicating that the delta marker time is 9.52 symbols.

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: EVM versus Time

Group Calculate commands

Syntax CALCulate:EVM:MARKer<x>:DELTa:Y?

Related Commands CALCulate:EVM:MARKer<x>:DELTa:X?

Arguments None

Returns <NRf> Delta marker amplitude for the selected marker in percent (%).

Examples CALCULATE:EVM:MARKER1:DELTA:Y? might return 1.62, indicating that the delta marker amplitude is 1.62%.

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: EVM versus Time

Group Calculate commands

Syntax `CALCulate:EVM:MARKer<x>:MAXimum`

Arguments None

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

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: EVM versus Time

Group Calculate commands

Syntax `CALCulate:EVM:MARKer<x>:PEAK:HIGHer`

Related Commands CALCulate:EVM:MARKer<x>:PEAK:LOWer

Arguments None

Examples `CALCULATE:EVM:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: EVM versus Time

Group Calculate commands

Syntax `CALCulate:EVM:MARKer<x>:PEAK:LEFT`

Related Commands CALCulate:EVM:MARKer<x>:PEAK:RIGHt

Arguments None

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

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: EVM versus Time

Group Calculate commands

Syntax `CALCulate:EVM:MARKer<x>:PEAK:LOWer`

Related Commands CALCulate:EVM:MARKer<x>:PEAK:HIGHer

Arguments None

Examples `CALCULATE:EVM:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: EVM versus Time

Group Calculate commands

Syntax `CALCulate:EVM:MARKer<x>:PEAK:RIGHt`

Related Commands CALCulate:EVM:MARKer<x>:PEAK:LEFT

Arguments None

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

CALCulate:EVM:MARKer<x>:X

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: EVM versus Time

Group Calculate commands

Syntax

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

Arguments

<value>::=<NRf> specifies the horizontal position of the marker.
Range: (analysis offset) to (analysis offset) + (analysis length).
The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples

`CALCULATE:EVM:MARKER1:X 1.5us` places Marker 1 (M1) at 1.5 µ s on the trace.

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

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

Conditions

Measurement views: EVM versus Time

Group

Calculate commands

Syntax

```
CALCulate:EVM:MARKer<x>:Y?
```

Related Commands

CALCulate:EVM:MARKer<x>:X

Arguments

None

Returns

<NRf> Marker amplitude of the selected marker in percent (%).

Examples

`CALCULATE:EVM:MARKER1:Y?` might return `5.34`, indicating Marker 1 (M1) is at 5.34%.

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

Returns the delta marker time for the selected marker on the Frequency Deviation vs Time trace.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Frequency Deviation vs Time

Group Calculate commands

Syntax `CALCulate:FDVTime:MARKer<x>:DELTa:X[:TIME]?`

Related Commands CALCulate:FDVTime:MARKer<x>:DELTa:Y?

Arguments None

Returns `<NRf>` Delta marker time for the selected marker.
Use the [SENSe]:DDEMod:TIMe:UNITs command to select the time unit: symbols (default) or seconds.

Examples `CALCULATE:FDVTIME:MARKER1:DELTA:X:TIME?` might return `62.75`, indicating that the delta marker time is 62.75 symbols.

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

Returns the delta marker amplitude for the selected marker on the Frequency Deviation vs Time trace.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Frequency Deviation vs Time

Group Calculate commands

Syntax `CALCulate:FDVTime:MARKer<x>:DELTa:Y?`

Related Commands CALCulate:FDVTime:MARKer<x>:DELTa:X[:TIME]?

Arguments None

Returns `<NRf>` Delta marker amplitude for the selected marker in Hz.

Examples `CALCULATE:FDVTIME:MARKER1:DELTA:Y?` might return `-563.7E+3`, indicating that the delta marker amplitude is -563.7 kHz.

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

Moves the selected marker to the highest peak on the Frequency Deviation vs Time trace.

Conditions Measurement views: Frequency Deviation vs Time

Group Calculate commands

Syntax `CALCulate:FDVTime:MARKer<x>:MAXimum`

Arguments None

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

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

Moves the selected marker to the next peak higher in amplitude on the Frequency Deviation vs Time trace.

Conditions Measurement views: Frequency Deviation versus Time

Group Calculate commands

Syntax `CALCulate:FDVTime:MARKer<x>:PEAK:HIGHer`

Related Commands CALCulate:FDVTime:MARKer<x>:PEAK:LOWer

Arguments None

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

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

Moves the selected marker to the next peak to the left on the Frequency Deviation vs Time trace.

Conditions Measurement views: Frequency deviation versus Time

Group Calculate commands

Syntax `CALCulate:FDVTime:MARKer<x>:PEAK:LEFT`

Related Commands CALCulate:FDVTime:MARKer<x>:PEAK:RIGHt

Arguments None

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

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

Moves the selected marker to the next peak lower in amplitude on the Frequency deviation vs Time trace.

Conditions Measurement views: Frequency Deviation vs Time

Group Calculate commands

Syntax `CALCulate:FDVTime:MARKer<x>:PEAK:LOWer`

Related Commands CALCulate:FDVTime:MARKer<x>:PEAK:HIGHer

Arguments None

Examples `CALCULATE:FDVTIME:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

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

Moves the selected marker to the next peak to the right on the trace in the Frequency Deviation vs Time measurement.

Conditions Measurement views: Frequency Deviation vs Time

Group Calculate commands

Syntax `CALCulate:FDVTime:MARKer<x>:PEAK:RIGHt`

Arguments None

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

CALCulate:FDVTime:MARKer<x>:X[:TIME]

Sets or queries the horizontal position (time) of the selected marker in the Frequency Deviation vs Time measurement.

Conditions Measurement views: Frequency Deviation vs Time

Group Calculate commands

Syntax
```
CALCulate:FDVTime:MARKer<x>:X[:TIME] <value>
CALCulate:FDVTime:MARKer<x>:X[:TIME]?
```

Related Commands CALCulate:FDVTime:MARKer<x>:Y?

Arguments `<value>::=<NRf>` specifies the horizontal position of the marker.
Range: (analysis offset) to [(analysis offset) + (analysis length)].
Use the [SENSe]:DDEMod:TIMe:UNITs command to select the time unit: symbols (default) or seconds.

Examples `CALCULATE:FDVTIME:MARKER1:X:TIME 38.5` places Marker 1 (M1) at 38.5 symbols on the trace.

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

Queries the marker amplitude of the selected marker in the Frequency Deviation vs Time measurement.

Conditions Measurement views: Frequency Deviation vs Time

Group Calculate commands

Syntax `CALCulate:FDVTime:MARKer<x>:Y?`

Related Commands CALCulate:FDVTime:MARKer<x>:X[:TIME]

Arguments None

Returns `<NRf>` Marker amplitude of the selected marker in Hz.

Examples `CALCULATE:FDVTIME:MARKER1:Y?` might return `-15.34E+6`, indicating Marker 1 (M1) is at -15.34 MHz.

CALCulate:{FSETtling|PSETtling}:MARKer<x>:DELTa:X? (Query Only)

Returns the delta marker time for the selected marker in the Frequency and Phase Settling displays.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Frequency and Phase Settling

Group Calculate commands

Syntax `CALCulate:{FSETtling|PSETtling}:MARKer<x>:DELTa:X?`

Related Commands CALCulate:FVTime:MARKer<x>:DELTa:Y?

Arguments None

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

Examples CALCULATE:FSETTLING:MARKER1:DELTA:X? might return 120.0E-9, indicating that the delta marker time is 120 ns.

CALCulate:{FSETtling|PSETtling}:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker frequency for the selected marker in the Frequency and Phase Settling displays.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Frequency and Phase Settling

Group Calculate commands

Syntax CALCulate:{FSETtling|PSETtling}:MARKer<x>:DELTa:Y?

Related Commands CALCulate:{FSETtling|PSETtling}:MARKer<x>:DELTa:X?

Arguments None

Returns <NRf> Delta marker frequency for the selected marker in Hz.

Examples CALCULATE:FSETTLING:MARKER1:DELTA:Y? might return 27.05E+3, indicating that the delta marker frequency is 27.05 kHz.

CALCulate:{FSETtling|PSETtling}:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the trace in the Frequency or Phase Settling display.

Conditions Measurement views: Frequency and Phase Settling

Group Calculate commands

Syntax CALCulate:{FSETtling|PSETtling}:MARKer<x>:MAXimum

Arguments None

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

CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on the Frequency or Phase Settling trace.

Conditions Measurement views: Frequency and Phase Settling

Group Calculate commands

Syntax CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:HIGHer

Related Commands CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:LOWer

Arguments None

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

CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the trace in the Frequency or Phase Settling display.

Conditions Measurement views: Frequency and Phase Settling

Group Calculate commands

Syntax CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:LEFT

Related Commands CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:RIGHt

Arguments None

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

CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the Frequency or Phase Settling trace.

Conditions Measurement views: Frequency and Phase Settling

Group Calculate commands

Syntax `CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:LOWer`

Related Commands CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:HIGHer

Arguments None

Examples `CALCULATE:FSETTLING:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the trace in the Frequency or Phase Settling display.

Conditions Measurement views: Frequency versus Time

Group Calculate commands

Syntax `CALCulate:{FSETtling|PSETtling}:MARKer<x>:PEAK:RIGHt`

Related Commands CALCulate:FVTime:MARKer<x>:PEAK:LEFT

Arguments None

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

CALCulate:{FSETtling|PSETtling}:MARKer<x>:TRACe

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

Conditions Measurement views: Frequency and Phase Settling

Group Calculate commands

Syntax
```
CALCulate:{FSETtling|PSETtling}:MARKer<x>:TRACe { TRACE1 | TRACE2 }
CALCulate:{FSETtling|PSETtling}:MARKer<x>:TRACe?
```

Arguments `TRACE1` places the specified marker on Trace 1.

`TRACE2` places the specified marker on Trace 2.

Examples `CALCULATE:FSETTLING:MARKER1:TRACE TRACE1` places Marker 1 (M1) on Trace 1.

CALCulate:{FSETtling|PSETtling}:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the Frequency and Phase Settling display.

Conditions Measurement views: Frequency and Phase Settling

Group Calculate commands

Syntax
```
CALCulate:{FSETtling|PSETtling}:MARKer<x>:X <value>
CALCulate:{FSETtling|PSETtling}:MARKer<x>:X?
```

Related Commands CALCulate:{FSETtling|PSETtling}:MARKer<x>:Y?

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

Examples CALCULATE:FVTIME:MARKER1:X 1.5u places Marker 1 (M1) at 1.5 µs on the trace.

CALCulate:{FSETtling|PSETtling}:MARKer<x>:Y? (Query Only)

Queries the marker amplitude of the selected marker in the Frequency and Phase Settling display.

Conditions Measurement views: Frequency and Phase Settling

Group Calculate commands

Syntax CALCulate:{FSETtling|PSETtling}:MARKer<x>:Y?

Related Commands CALCulate:{FSETtling|PSETtling}:MARKer<x>:X

Arguments None

Returns <NRf> Marker amplitude of the selected marker in Hz or degrees.

Examples CALCULATE:FSETTLING:MARKER1:Y? might return -15.34E+6, indicating Marker 1 (M1) is at -15.34 MHz.

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Frequency versus Time

Group Calculate commands

Syntax `CALCulate:FVTime:MARKer<x>:DELTa:X?`

Related Commands CALCulate:FVTime:MARKer<x>:DELTa:Y?

Arguments None

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

Examples `CALCULATE:FVTIME:MARKER1:DELTA:X?` might return `120.0E-9`, indicating that the delta marker time is 120 ns.

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

Returns the delta marker frequency for the selected marker in the Frequency versus Time measurement.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Frequency versus Time

Group Calculate commands

Syntax `CALCulate:FVTime:MARKer<x>:DELTa:Y?`

Related Commands CALCulate:FVTime:MARKer<x>:DELTa:X?

Arguments None

Returns `<NRf>` Delta marker frequency for the selected marker in Hz.

Examples `CALCULATE:FVTIME:MARKER1:DELTA:Y?` might return `27.05E+3`, indicating that the delta marker frequency is 27.05 kHz.

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

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

Conditions Measurement views: Frequency versus Time

Group Calculate commands

Syntax `CALCulate:FVTime:MARKer<x>:MAXimum`

Arguments None

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

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

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

Conditions Measurement views: Frequency versus Time

Group Calculate commands

Syntax `CALCulate:FVTime:MARKer<x>:PEAK:HIGHer`

Related Commands CALCulate:FVTime:MARKer<x>:PEAK:LOWer

Arguments None

Examples `CALCULATE:FVTIME:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

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

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

Conditions Measurement views: Frequency versus Time

Group Calculate commands

Syntax `CALCulate:FVTime:MARKer<x>:PEAK:LEFT`

Related Commands CALCulate:FVTime:MARKer<x>:PEAK:RIGHt

Arguments None

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

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

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

Conditions Measurement views: Frequency versus Time

Group Calculate commands

Syntax `CALCulate:FVTime:MARKer<x>:PEAK:LOWer`

Related Commands CALCulate:FVTime:MARKer<x>:PEAK:HIGHer

Arguments None

Examples `CALCULATE:FVTIME:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

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

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

Conditions Measurement views: Frequency versus Time

Group Calculate commands

Syntax `CALCulate:FVTime:MARKer<x>:PEAK:RIGHt`

Related Commands CALCulate:FVTime:MARKer<x>:PEAK:LEFT

Arguments None

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

CALCulate:FVTime:MARKer<x>:X

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

Conditions Measurement views: Frequency versus Time

Group Calculate commands

Syntax
```
CALCulate:FVTime:MARKer<x>:X <value>
CALCulate:FVTime:MARKer<x>:X?
```

Related Commands CALCulate:FVTime:MARKer<x>:Y?

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

Examples `CALCULATE:FVTIME:MARKER1:X 1.5u` places Marker 1 (M1) at 1.5 µs on the trace.

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

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

Conditions Measurement views: Frequency versus Time

Group Calculate commands

Syntax `CALCulate:FVTime:MARKer<x>:Y?`

Related Commands CALCulate:FVTime:MARKer<x>:X

Arguments None

Returns `<NRf>` Marker amplitude of the selected marker in Hz.

Examples `CALCULATE:FVTIME:MARKER1:Y?` might return `-15.34E+6`, indicating Marker 1 (M1) is at -15.34 MHz.

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

Returns the delta marker time for the selected marker in the RF I&Q versus Time measurement.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: RF I&Q versus Time

Group Calculate commands

Syntax `CALCulate:IQVTime:MARKer<x>:DELTa:X?`

Related Commands CALCulate:IQVTime:MARKer<x>:DELTa:Y?

Arguments None

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

Examples CALCULATE:IQVTIME:MARKER1:DELTA:X? might return 120.0E-9, indicating that the delta marker time is 120 ns.

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

Returns the delta marker amplitude for the selected marker in the RF I&Q versus Time measurement.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: RF I&Q versus Time

Group Calculate commands

Syntax CALCulate:IQVTime:MARKer<x>:DELTa:Y?

Related Commands CALCulate:IQVTime:MARKer<x>:DELTa:X?

Arguments None

Returns <NRf> Delta marker amplitude for the selected marker in volts.

Examples CALCULATE:IQVTIME:MARKER1:DELTA:Y? might return -3.45E-3, indicating that the delta marker amplitude is -3.45 mV.

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

Moves the selected marker to the highest peak on the trace in the RF I&Q versus Time measurement.

Conditions Measurement views: RF I&Q versus Time

Group Calculate commands

Syntax `CALCulate:IQVTime:MARKer<x>:MAXimum`

Arguments None

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

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

Moves the selected marker to the next peak higher in amplitude on the RF I&Q versus Time trace.

Conditions Measurement views: RF I&Q versus Time

Group Calculate commands

Syntax `CALCulate:IQVTime:MARKer<x>:PEAK:HIGHer`

Related Commands CALCulate:IQVTime:MARKer<x>:PEAK:LOWer

Arguments None

Examples `CALCULATE:IQVTIME:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

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

Moves the selected marker to the next peak to the left on the trace in the RF I&Q versus Time measurement.

Conditions Measurement views: RF I&Q versus Time

Group Calculate commands

Syntax `CALCulate:IQVTime:MARKer<x>:PEAK:LEFT`

Related Commands CALCulate:IQVTime:MARKer<x>:PEAK:RIGHt

Arguments None

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

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

Moves the selected marker to the next peak lower in amplitude on the RF I&Q versus Time trace.

Conditions Measurement views: RF I&Q versus Time

Group Calculate commands

Syntax `CALCulate:IQVTime:MARKer<x>:PEAK:LOWer`

Related Commands CALCulate:IQVTime:MARKer<x>:PEAK:HIGHer

Arguments None

Examples `CALCULATE:IQVTIME:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

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

Moves the selected marker to the next peak to the right on the trace in the RF I&Q versus Time measurement.

Conditions Measurement views: RF I&Q versus Time

Group Calculate commands

Syntax `CALCulate:IQVTime:MARKer<x>:PEAK:RIGHt`

Related Commands CALCulate:IQVTime:MARKer<x>:PEAK:LEFT

Arguments None

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

CALCulate:IQVTime:MARKer<x>:TRACe

Places the selected marker on the I or Q trace in the RF I&Q versus Time measurement. The query command returns which trace the selected marker is placed on.

Conditions Measurement views: RF I&Q versus Time

Group Calculate commands

Syntax
```
CALCulate:IQVTime:MARKer<x>:TRACe { TRACE1 | TRACE2 }
CALCulate:IQVTime:MARKer<x>:TRACe?
```

Arguments `TRACE1` places the selected marker on the I trace.

`TRACE2` places the selected marker on the Q trace.

Examples `CALCULATE:IQVTIME:MARKER1:TRACE TRACE1` places Marker 1 (M1) on the I trace.

CALCulate:IQVTime:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the RF I&Q versus Time measurement.

Conditions Measurement views: RF I&Q versus Time

Group Calculate commands

Syntax
```
CALCulate:IQVTime:MARKer<x>:X <value>
CALCulate:IQVTime:MARKer<x>:X?
```

Related Commands CALCulate:IQVTime:MARKer<x>:Y?

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

Examples CALCULATE:IQVTIME:MARKER1:X 1.5us places Marker 1 (M1) at 1.5 µs on the trace.

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

Queries the marker amplitude of the selected marker in the RF I&Q versus Time measurement.

Conditions Measurement views: RF I&Q versus Time

Group Calculate commands

Syntax CALCulate:IQVTime:MARKer<x>:Y?

Related Commands CALCulate:IQVTime:MARKer<x>:X

Arguments None

Returns <NRf> Marker amplitude of the selected marker in volts.

Examples CALCULATE:IQVTIME:MARKER1:Y? might return 25.803E-3, indicating Marker 1 (M1) is at 25.803 mV.

CALCulate:LTE:ACLR:MARKer:<x>PEAK:HIGHer (No Query Form)

Moves the specified marker to the next peak higher in amplitude on the trace in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Calculate commands

Syntax CALCulate:LTE:ACLR:MARKer:<x>PEAK:HIGHer

Arguments None

Examples CALCULATE:LTE:ACLR:MARKER1:PEAK:HIGHER moves marker 1 (M1) to the next peak higher in amplitude on the trace.

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

Returns the frequency of the specified delta marker on the trace in the LTE ACLR display. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement view: LTE ACLR

<x> parameters: 1 to 4; MARKer0 (reference marker) is invalid.

Group Calculate commands

Syntax CALCulate:LTE:ACLR:MARKer<x>:DELTa:X?

Returns <NR3>

Examples CALCULATE:LTE:ACLR:MARKER1:DELTA:X? might return 1.25000000E+6, indicating that the frequency of the specified delta marker is 1.25 MHz.

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

Returns the amplitude for the specified delta marker on the trace in the LTE ACLR display. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement view: LTE ACLR

<x> parameters: 1 to 4; MARKer0 (reference marker) is invalid.

Group Calculate commands

Syntax CALCulate:LTE:ACLR:MARKer<x>:DELTa:Y?

Returns <NRf>

Examples CALC:LTEACLR:MARK1:DELT:Y? might return 23.45, indicating that the amplitude of the specified delta marker is 23.45 dB.

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

Positions the specified marker at the highest peak on the trace in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

<x> parameters: 0 to 4

Group Calculate commands

Syntax CALCulate:LTE:ACLR:MARKer<x>:MAXimum

Arguments None

Examples CALCULATE:LTE:ACLR:MARKER1:MAXIMUM moves marker 1 (M1) to the highest peak on the trace.

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

Moves the selected marker to the next peak to the left on the trace in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

<x> parameters: 0 to 4

Group Calculate commands

Syntax CALCulate:LTE:ACLR:MARKer<x>:PEAK:LEFT

Arguments None

Examples `CALCULATE:LTE:ACLR:MARKER1:PEAK:LEFT` moves marker 1 (M1) to the next peak to the left on the trace.

CALCulate:LTE:ACLR:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the trace in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

<x> parameters: 0 to 4

Group Calculate commands

Syntax `CALCulate:LTE:ACLR:MARKer<x>:PEAK:LOWer`

Arguments None

Examples `CALCULATE:LTE:ACLR:MARKER1:PEAK:LOWER` moves marker 1 (M1) to the next peak lower in amplitude on the trace.

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

Moves the selected marker to the next peak to the right on the trace in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

<x> parameters: 0 to 4

Group Calculate commands

Syntax `CALCulate:LTE:ACLR:MARKer<x>:PEAK:RIGHt`

Arguments None

Examples `CALCULATE:LTE:ACLR:MARKER1:PEAK:RIGHT` moves marker 1 (M1) to the next peak to the right on the trace.

CALCulate:LTE:ACLR:MARKer<x>:SET:CENTer (No Query Form)

Moves the specified marker to the center frequency in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

<x> parameters: 0 to 4

Group Calculate commands

Syntax `CALCulate:LTE:ACLR:MARKer<x>:SET:CENTer`

Arguments None

Examples `CALCULATE:LTE:ACLR:MARKER1:SET:CENTER` moves marker 1 (M1) to the center frequency on the trace.

CALCulate:LTE:ACLR:MARKer<x>:X

Sets or queries the horizontal position of the selected marker on the trace in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

<x> parameters: 0 to 4

Group Calculate commands

Syntax
```
CALCulate:LTE:ACLR:MARKer<x>:X <value>
CALCulate:LTE:ACLR:MARKer<x>:X?
```

Arguments <value>::=<NRf> specifies the horizontal position of the marker.

Returns <NRf>

Examples CALCULATE:LTE:ACLR:MARKER1:X 800.000000000E+6 places marker 1 (M1) at 800 MHz on the trace.

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

Queries the amplitude of the selected marker on the trace in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

<x> parameters: 0 to 4

Group Calculate commands

Syntax CALCulate:LTE:ACLR:MARKer<x>:Y?

Returns <NRf>

Examples CALCULATE:LTE:ACLR:MARKER1:Y? might return –34.28, indicating that the amplitude of marker 1 (M1) is –34.28 dBm.

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

Returns the frequency of the specified delta marker on the trace in the LTE Channel Spectrum display. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement view: LTE Channel Spectrum

<x> parameters: 1 to 4; MARKer0 (reference marker) is invalid.

Group Calculate commands

Syntax CALCulate:LTE:CHSPectrum:MARKer<x>:DELTa:X?

Returns <NR3>

Examples CALCULATE:LTE:CHSPECTRUM:MARKER1:DELTA:X? might return 1.25000000E+6, indicating that the frequency of the specified delta marker is 1.25 MHz.

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

Returns the amplitude for the specified delta marker on the trace in the LTE Channel Spectrum display. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement view: LTE Channel Spectrum

<x> parameters: 1 to 4; MARKer0 (reference marker) is invalid.

Group Calculate commands

Syntax CALCulate:LTE:CHSPectrum:MARKer<x>:DELTa:Y?

Returns <NRf>

Examples CALC:LTE:CHSPECTRUM:MARK1:DELT:Y? might return 23.45, indicating that the amplitude of the specified delta marker is 23.45 dB.

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

Positions the marker at the highest peak on the trace in the LTE Channel Spectrum display.

Conditions Measurement view: LTE Channel Spectrum

<x> parameters: 0 to 4

Group Calculate commands

Syntax CALCulate:LTE:CHSPectrum:MARKer<x>:MAXimum

Arguments None

Examples `CALCULATE:LTE:CHSPECTRUM:MARKER1:MAXIMUM` moves marker 1 (M1) to the highest peak on the trace.

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

Moves the selected marker to the next peak to the left on the trace in the LTE Channel Spectrum display.

Conditions Measurement view: LTE Channel Spectrum

<x> parameters: 0 to 4

Group Calculate commands

Syntax `CALCulate:LTE:CHSPectrum:MARKer<x>:PEAK:LEFT`

Arguments None

Examples `CALCULATE:LTE:CHSPECTRUM:MARKER1:PEAK:LEFT` moves marker 1 (M1) to the next peak to the left on the trace.

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

Moves the selected marker to the next peak to the right on the trace in the LTE Channel Spectrum display.

Conditions Measurement view: LTE Channel Spectrum

<x> parameters: 0 to 4

Group Calculate commands

Syntax `CALCulate:LTE:CHSPectrum:MARKer<x>:PEAK:RIGHt`

Arguments None

Examples `CALCULATE:LTE:CHSPECTRUM:MARKER1:PEAK:RIGHT` moves marker 1 (M1) to the next peak to the right on the trace.

CALCulate:LTE:CHSPectrum:MARKer<x>:X

Sets or queries the horizontal position of the selected marker on the trace in the LTE Channel Spectrum display.

Conditions Measurement view: LTE Channel Spectrum

<x> parameters: 0 to 4

Group Calculate commands

Syntax

```
CALCulate:LTE:CHSPectrum:MARKer<x>:X <value>
CALCulate:LTE:CHSPectrum:MARKer<x>:X?
```

Arguments <value>::=<NRf> specifies the horizontal position of the marker.

Returns <NRf>

Examples `CALCULATE:LTE:CHSPECTRUM:MARKER1:X 800.000000000E+6` places marker 1 (M1) at 800 MHz on the trace.

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

Queries the amplitude of the selected marker on the trace in the LTE Channel Spectrum display.

Conditions Measurement view: LTE Channel Spectrum

<x> parameters: 0 to 4

Group Calculate commands

Syntax

```
CALCulate:LTE:CHSPectrum:MARKer<x>:Y?
```

Returns <NRf>

Examples `CALCULATE:LTE:CHSPECTRUM:MARKER1:Y?` might return –34.28, indicating that the amplitude of marker 1 (M1) is –34.28 dBm.

CALCulate:LTE:CONSte:MARKer<x>:FREQuency

Sets or returns the subcarrier number of the LTE symbol in the LTE Constellation display. This constellation shows only primary and secondary synchronization signals. Any input outside this range will show the marker set to -31 or +31.

Conditions Measurement view: LTE Constellation

<x> parameters: 0 to 4

Range: -31 to +31

Group Calculate commands

Syntax
```
CALCulate:LTE:CONSte:MARKer<x>:FREQuency <value>
CALCulate:LTE:CONSte:MARKer<x>:FREQuency?
```

Arguments <value>::<NRf> specifies the subcarrier number of the specified marker

Returns <NRf> The value type depends on where the marker is placed.

Examples `CALCULATE:LTE:CONSTE:MARKER2:FREQUENCY?` might return +31, indicating that Marker 2 (M2) is positioned at the subcarrier +31 in the LTE Constellation display.

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

Queries the marker magnitude of the selected marker in the LTE Constellation display.

Conditions Measurement view: LTE Constellation

<x> parameters: 0 to 4

Group Calculate commands

Syntax
```
CALCulate:LTE:CONSte:MARKer<x>:MAGNitude?
```

Returns <NRf> The magnitude for the selected marker.

Examples `CALCULATE:LTE:CONSTE:MARKER2:MAGNITUDE` might return "1.004" indicating readout of Marker 2 (M2) is "1.004".

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

Queries the phase readout of the selected marker in the LTE Constellation display.

Conditions Measurement view: LTE Constellation

<x> parameters: 0 to 4

Group Calculate commands

Syntax `CALCulate:LTE:CONSte:MARKer<x>:PHASe?`

Returns <NRf> The phase readout for the selected marker.

Examples `CALCULATE:LTE:CONSTE:MARKER1:PHASE?` might return 35.74, indicating the phase readout of Marker 1 (M1) is 35.74°.

CALCulate:LTE:CONSte:MARKer<x>:TIME

Sets or queries the X position (Symbol Value) of the selected marker in the LTE Constellation display.

Conditions Measurement view: LTE Constellation

<x> parameters: 0 to 4

For FDD, the range is 0 and 1.

For TDD the range is 0 and 3.

Group Calculate commands

Syntax
```
CALCulate:LTE:CONSte:MARKer<x>:TIME <value>
CALCulate:LTE:CONSte:MARKer<x>:TIME?
```

Arguments <value>::=<NRf> specifies the specified marker symbol number.

Returns <NR1>

Examples `CALCULATE:LTE:CONSTE:MARKER<x>:TIME 0` places Marker 1 (M1) at the symbol 0 on range for time.

CALCulate:LTE:CONSte:MARKer<x>:TYPE? (Query Only)

Queries the data type of the selected marker in LTE Constellation display. The data type can either be “PSS” (Primary Synchronization Signal) or “SSS” (Secondary Synchronization Signal).

Conditions Measurement view: LTE Constellation

<x> parameters: 0 to 4

Group Calculate commands

Syntax `CALCulate:LTE:CONSte:MARKer<x>:TYPE?`

Returns PSS means that the data type for the selected marker is Primary Synchronization Signal

SSS means that the data type for the selected marker is Secondary Synchronization Signal

Examples `CALCULATE:LTE:CONSTE:MARKER1:TYPE?` might return SSS, indicating that the data type for the Marker 1 (M1) is SSS.

CALCulate:LTE:PVTime:MARKer:<x>PEAK:HIGHer (No Query Form)

Moves the specified marker to the next peak higher in amplitude on the trace in the LTE Power vs Time display.

Conditions Measurement view: LTE Power vs Time

Parameter <x>: 0 to 4

Group Calculate commands

Syntax `CALCulate:LTE:PVTime:MARKer:<x>PEAK:HIGHer`

Arguments None

Examples `CALCULATE:LTE:PVTIME:MARKER1:PEAK:HIGHER` moves marker 1 (M1) to the next peak higher in amplitude on the trace.

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

Returns the frequency of the specified delta marker on the trace in the LTE Power vs Time display. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Conditions Measurement view: LTE Power vs Time

<x> parameters: 1, 2, 3...(valid integral values range from 1 to the number of markers active for the display); MARKer0 (reference marker) is invalid.

Querying an out-of-range value causes an execution error (-222, "Data out of range").

Group Calculate commands

Syntax `CALCulate:LTE:PVTime:MARKer<x>:DELTa:X?`

Returns <NR3>

Examples `CALCULATE:LTE:PVTIME:MARKER1:DELTA:X?` might return 1.28000000E-6, indicating that the frequency of the specified delta marker is 1.28 µs.

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

Returns the amplitude for the specified delta marker on the trace in the LTE Power vs Time display using the currently selected amplitude unit. You can set/query the amplitude unit with the `SENSe:POWer:UNITs` command. The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Conditions Measurement view: LTE Power vs Time

<x> parameters: 1, 2, 3...(valid integral values range from 1 to the number of markers active for the display); MARKer0 (reference marker) is invalid.

Querying an out-of-range value causes an execution error (-222, “Data out of range”).

Group Calculate commands

Syntax `CALCulate:LTE:PVTime:MARKer<x>:DELTa:Y?`

Returns <NRf>

Examples `CALCULATE:LTE:PVTIME:MARKER3:DELTA:Y?` might return –12.91, indicating that the amplitude of the delta marker 3 (M3) is –12.91 dB.

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

Positions the specified marker at the highest peak on the trace in the LTE Power vs Time display.

Conditions Measurement view: LTE Power vs Time

<x> parameters: 0 to 4

Group Calculate commands

Syntax `CALCulate:LTE:PVTime:MARKer<x>:MAXimum`

Arguments None

Examples `CALCULATE:LTE:PVTIME:MARKER1:MAXIMUM` moves marker 1 (M1) to the highest peak on the trace.

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

Moves the selected marker to the next peak to the left on the trace in the LTE Power vs Time display.

Conditions Measurement view: LTE Power vs Time

<x> parameters: 0 to 4

Group Calculate commands

Syntax `CALCulate:LTE:PVTime:MARKer<x>:PEAK:LEFT`

Arguments None

Examples `CALCULATE:LTE:PVTIME:MARKER1:PEAK:LEFT` moves marker 1 (M1) to the next peak to the left on the trace.

CALCulate:LTE:PVTime:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the trace in the LTE Power vs Time display.

Conditions Measurement view: LTE Power vs Time

<x> parameters: 0 to 4

Group Calculate commands

Syntax `CALCulate:LTE:PVTime:MARKer<x>:PEAK:LOWer`

Arguments None

Examples `CALCULATE:LTE:PVTIME:MARKER1:PEAK:LOWER` moves marker 1 (M1) to the next peak lower in amplitude on the trace.

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

Moves the selected marker to the next peak to the right on the trace in the LTE Power vs Time display.

Conditions Measurement view: LTE Power vs Time

<x> parameters: 0 to 4

Group Calculate commands

Syntax `CALCulate:LTE:PVTime:MARKer<x>:PEAK:RIGHt`

Arguments None

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

CALCulate:LTE:PVTime:MARKer<x>:X

Sets or queries the horizontal position of the selected marker on the trace in the LTE Power vs Time display.

Conditions Measurement view: LTE Power vs Time

<x> parameters: 0 to 4

Range: Start to Stop time (left to right edge of the horizontal axis). Using an out-of-range value causes an execution error (-222, "Data out of range").

Group Calculate commands

Syntax
```
CALCulate:LTE:PVTime:MARKer<x>:X <NRf>
CALCulate:LTE:PVTime:MARKer<x>:X?
```

Arguments <NRf> specifies the value of the horizontal position of the marker.

Returns <NRf>

Examples `CALCULATE:LTE:PVTIME:MARKER3:X 1.000000000E-6` places Marker 3(M3) at 1 µs on the trace.

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

Queries the amplitude of the selected marker on the trace in the LTE Power vs Time display. You can set/query the amplitude units with the `SENSe:POWer:UNITs` command.

Conditions Measurement view: LTE Power vs Time

<x> parameters: 0 to 4

Querying an out-of-range value causes an execution error (-222, “Data out of range”).

Group Calculate commands

Syntax `CALCulate:LTE:PVTime:MARKer<x>:Y?`

Returns <NRf> specifies the value of the horizontal position of the marker.

Examples `CALCULATE:LTE:PVTIME:MARKER1:Y?` might return 33.393856E-3, indicating that the amplitude of marker 1 (M1) is 0.033 dBm.

CALCulate:MARKer:ADD (No Query Form)

Adds a marker. Every execution of this command adds a marker from MR, then M1 to M4, sequentially.

NOTE. *If all markers are already turned on, the error message "Cannot add another marker" (execution error -200) is returned.*

Conditions Measurement views: All

Group Calculate commands

Syntax `CALCulate:MARKer:ADD`

Related Commands CALCulate:MARKer:AOFF, CALCulate:MARKer:DELete

Arguments None

Examples `CALCULATE:MARKER:ADD` adds a marker.

CALCulate:MARKer:AOFF (No Query Form)

Turns off all markers.

Conditions Measurement views: All

Group Calculate commands

Syntax `CALCulate:MARKer:AOFF`

Related Commands CALCulate:MARKer:ADD, CALCulate:MARKer:DELete

Arguments None

Examples `CALCULATE:MARKER:AOFF` turns off all markers.

CALCulate:MARKer:DELete (No Query Form)

Deletes the last marker added.

NOTE. *If all markers are turned off, the error message "Cannot delete another marker" (execution error -200) is returned.*

Conditions Measurement views: All

Group Calculate commands

Syntax `CALCulate:MARKer:DELete`

Related Commands CALCulate:MARKer:ADD, CALCulate:MARKer:AOFF

Arguments None

Examples `CALCULATE:MARKER:DELETE` deletes the last marker added.

CALCulate:MARKer:DENSity:EXCursion

Sets or queries the minimum excursion of DPX signal density, or how far the density (hit count for pixels) must be above the surrounding noise to be detected as a peak. This command is effective for the DPX bitmap trace.

For marker peak up and marker peak down operations, the signal is considered to be a peak if it exceeds the excursion and the threshold set by the CALCulate:MARKer:DENSity:THReshold command.

For marker peak left and marker peak right operations, the signal is considered to be a peak if it first exceeds the amplitude excursion set by CALCulate:MARKer:PEAK:EXCursion, the amplitude threshold set by CALCulate:MARKer:PEAK:THReshold, the density excursion set by CALCulate:MARKer:DENSity:EXCursion and the density threshold set by CALCulate:MARKer:DENSity:THReshold.

Conditions Measurement views: All

Group Calculate commands

Syntax
```
CALCulate:MARKer:DENSity:EXCursion <number>
CALCulate:MARKer:DENSity:EXCursion?
```

Related Commands CALCulate:MARKer:DENSity:THReshold, CALCulate:MARKer:PEAK:EXCursion, CALCulate:MARKer:PEAK:THReshold

Arguments `<number>::=<NR1>` specifies the minimum excursion density. Range: 0 to 100%.

Examples `CALCULATE:MARKER:DENSITY:EXCURSION30` sets the minimum excursion density to 30.

CALCulate:MARKer:DENSity:SMOothing

Sets or queries the number of pixels squared for smoothing the DPX signal density. This command is effective for the DPX bitmap trace.

Conditions Measurement views: All

Group Calculate commands

Syntax
```
CALCulate:MARKer:DENSity:SMOothing <number>
CALCulate:MARKer:DENSity:SMOothing?
```

Arguments <number>::=<NR1> specifies the number of pixels squared for smoothing the DPX signal density. Range: 1 to 20.

Examples CALCULATE:MARKER:DENSITY:SMOOTHING5 sets the smoothing number to 5.

CALCulate:MARKer:DENSity:THReshold

Sets or queries the threshold of DPX signal density above which the density (hit count for pixels) is detected as a peak. This command is effective for the DPX bitmap trace.

Conditions Measurement views: All

Group Calculate commands

Syntax
```
CALCulate:MARKer:DENSity:THReshold <number>
CALCulate:MARKer:DENSity:THReshold?
```

Related Commands CALCulate:MARKer:DENSity:EXCursion

Arguments <number>::=<NR1> specifies the threshold density for detecting peaks. Range: 0 to 100%.

Examples CALCULATE:MARKER:DENSITY:THRESHOLD300 sets the threshold density to 300.

CALCulate:MARKer:DRAG:SEARch:STATe

Enables or disables peak searching when dragging a marker in a plot.

Conditions Measurement views: All

Group Calculate commands

Syntax
```
CALCulate:MARKer:DRAG:SEARch:STATe { OFF | ON | 0 | 1 }
CALCulate:MARKer:DRAG:SEARch:STATe?
```

Arguments OFF or 0 disables peak searching in a plot.

ON or 1 enables peak searching in a plot.

Examples `CALCULATE:MARKER:DRAG:SEARCH:STATE OFF` disables peak searching when dragging a marker in a plot.

CALCulate:MARKer:MODE

Sets or queries the marker mode.

Conditions Measurement views: All

Group Calculate commands

Syntax
```
CALCulate:MARKer:MODE { ABSolute | DELTa }
CALCulate:MARKer:MODE?
```

Arguments `ABSolute` selects the absolute marker mode, in which the marker readout indicates the absolute value.

`DELTa` selects the delta marker mode, in which the marker readout indicates the relative value to the reference marker.

Examples `CALCULATE:MARKER:MODEDELTa` selects the delta marker mode.

CALCulate:MARKer:MODE

Sets or queries the marker mode.

Conditions Measurement views: All

Group Calculate commands

Syntax
```
CALCulate:MARKer:MODE { ABSolute | DELTa | POWer }
CALCulate:MARKer:MODE?
```

Arguments `ABSolute` selects the absolute marker mode, in which the marker readout indicates the absolute value.

DELTa selects the delta marker mode, in which the marker readout indicates the relative value to the selected marker.

POWer selects the power marker mode, in which the marker readout indicates the power value of the selected marker.

Examples CALCULATE:MARKER:MODE POWER selects the power marker mode.

CALCULATE:MARKER:MODE ? might return POW, meaning the Power is the assigned mode for the marker readout.

CALCulate:MARKer:PEAK:EXCursion

Sets or queries the minimum excursion level (how far a signal must be above the surrounding noise to be detected as a peak). The signal is considered to be a peak if it exceeds the minimum excursion level and the threshold level set by the CALCulate:MARKer:PEAK:THReshold command.

Conditions Measurement views: All

Group Calculate commands

Syntax
```
CALCulate:MARKer:PEAK:EXCursion <value>
CALCulate:MARKer:PEAK:EXCursion?
```

Arguments <value>::=<NRf> specifies the excursion level. Range: 0 to 100 dB.

Examples CALCULATE:MARKER:PEAK:EXCURSION10 sets the excursion level to 10 dB.

CALCulate:MARKer:PEAK:THReshold

Sets or queries the threshold level above which a signal is detected as a peak.

Conditions Measurement views: All

Group Calculate commands

Syntax
```
CALCulate:MARKer:PEAK:THReshold <value>
CALCulate:MARKer:PEAK:THReshold?
```

Arguments <value>::=<NRf> specifies the threshold level for detecting peaks. Range: -170 to +130 dBm.

Examples CALCULATE:MARKER:PEAK:THRESHOLD-50 sets the threshold level to -50 dBm.

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

Returns the delta marker frequency for the selected marker in the MCPR measurement.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: MCPR

Group Calculate commands

Syntax CALCulate:MCPower:MARKer<x>:DELTa:X?

Related Commands CALCulate:MCPower:MARKer<x>:DELTa:Y?

Arguments None

Returns <NRf> Delta marker frequency for the selected marker.

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

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

Returns the delta marker amplitude for the selected marker in the MCPR measurement.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: MCPR

Group Calculate commands

Syntax `CALCulate:MCPower:MARKer<x>:DELTa:Y?`

Related Commands CALCulate:MCPower:MARKer<x>:DELTa:X?

Arguments None

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

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

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

Moves the selected marker to the highest peak on the trace in the MCPR measurement.

Conditions Measurement views: MCPR

Group Calculate commands

Syntax `CALCulate:MCPower:MARKer<x>:MAXimum`

Arguments None

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

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

Moves the selected marker to the next peak to the left on the trace in the MCPR measurement.

Conditions Measurement views: MCPR

Group Calculate commands

Syntax `CALCulate:MCPower:MARKer<x>:PEAK:LEFT`

Related Commands CALCulate:MCPower:MARKer<x>:PEAK:RIGHt

Arguments None

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

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

Moves the selected marker to the next peak to the right on the trace in the MCPR measurement.

Conditions Measurement views: MCPR

Group Calculate commands

Syntax `CALCulate:MCPower:MARKer<x>:PEAK:RIGHt`

Related Commands CALCulate:MCPower:MARKer<x>:PEAK:LEFT

Arguments None

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

CALCulate:MCPower:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the MCPR measurement.

Conditions Measurement views: MCPR

Group Calculate commands

Syntax
```
CALCulate:MCPower:MARKer<x>:X <value>
CALCulate:MCPower:MARKer<x>:X?
```

Related Commands CALCulate:MCPower:MARKer<x>:Y?

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

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

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

Queries the marker amplitude of the selected marker in the MCPR measurement.

Conditions Measurement views: MCPR

Group Calculate commands

Syntax
```
CALCulate:MCPower:MARKer<x>:Y?
```

Related Commands CALCulate:MCPower:MARKer<x>:X

Arguments None

Returns <NRf> Marker amplitude of the selected marker.

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

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Magnitude error versus Time

Group Calculate commands

Syntax `CALCulate:MERRor:MARKer<x>:DELTa:X?`

Related Commands CALCulate:MERRor:MARKer<x>:DELTa:Y?

Arguments None

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

The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples `CALCULATE:MERROR:MARKER1:DELTA:X?` might return `9.52`, indicating that the delta marker time is 9.52 symbols.

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Magnitude error versus Time

Group Calculate commands

Syntax `CALCulate:MERRor:MARKer<x>:DELTa:Y?`

Related Commands CALCulate:MERRor:MARKer<x>:DELTa:X?

Arguments None

Returns <NRf> Delta marker amplitude for the selected marker in percent (%).

Examples CALCULATE:MERROR:MARKER1:DELTA:Y? might return 3.84, indicating that the delta marker amplitude is 3.84%.

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

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

Conditions Measurement views: Magnitude error versus Time

Group Calculate commands

Syntax CALCulate:MERRor:MARKer<x>:MAXimum

Arguments None

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

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

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

Conditions Measurement views: Magnitude error versus Time

Group Calculate commands

Syntax CALCulate:MERRor:MARKer<x>:PEAK:HIGHer

Related Commands CALCulate:MERRor:MARKer<x>:PEAK:LOWer

Arguments None

Examples `CALCULATE:MERROR:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

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

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

Conditions Measurement views: Magnitude error versus Time

Group Calculate commands

Syntax `CALCulate:MERRor:MARKer<x>:PEAK:LEFT`

Related Commands CALCulate:MERRor:MARKer<x>:PEAK:RIGHt

Arguments None

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

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

Moves the selected marker to the next peak lower in amplitude on the Magnitude error versus Time trace.

Conditions Measurement views: Magnitude error versus Time

Group Calculate commands

Syntax `CALCulate:MERRor:MARKer<x>:PEAK:LOWer`

Related Commands CALCulate:MERRor:MARKer<x>:PEAK:HIGHer

Arguments None

Examples `CALCULATE:MERROR:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

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

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

Conditions Measurement views: Magnitude error versus Time

Group Calculate commands

Syntax `CALCulate:MERRor:MARKer<x>:PEAK:RIGHt`

Related Commands CALCulate:MERRor:MARKer<x>:PEAK:LEFT

Arguments None

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

CALCulate:MERRor:MARKer<x>:X

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

Conditions Measurement views: Magnitude error versus Time

Group Calculate commands

Syntax
```
CALCulate:MERRor:MARKer<x>:X <value>
CALCulate:MERRor:MARKer<x>:X?
```

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

The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples `CALCULATE:MERROR:MARKER1:X 1.5u` places Marker 1 (M1) at 1.5 µs on the trace.

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

Queries the marker amplitude of the selected marker in the Magnitude error versus Time measurement.

Conditions Measurement views: Magnitude error versus Time

Group Calculate commands

Syntax `CALCulate:MERRor:MARKer<x>:Y?`

Related Commands CALCulate:MERRor:MARKer<x>:X

Arguments None

Returns `<NRf>` Marker amplitude of the selected marker in percent (%).

Examples `CALCULATE:MERROR:MARKER1:Y?` might return `5.34`, indicating Marker 1 (M1) is at 5.34%.

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

Queries the frequency difference between MR (reference marker) and the selected marker.

Conditions Measurement view: Noise Figure

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid.

Group Calculate commands

Syntax `CALCulate:NOISe:FIGure:MARKer<x>:DELTa:X?`

Related Commands CALCulate:NOISe:FIGure:MARKer<x>:Y?

Returns <NRf> is the frequency difference value between MR and the selected marker.

Examples `CALCULATE:NOISE:FIGURE:MARKER1:DELTA:X?` might return `0.0000`, indicating the frequency difference between MR (reference marker) and the selected marker (M1) is 0.0000 dB.

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

Queries the power difference value between MR (reference marker) and the selected marker.

Conditions Measurement view: Noise Figure

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid.

Group Calculate commands

Syntax `CALCulate:NOISe:FIGure:MARKer<x>:DELTa:Y?`

Related Commands CALCulate:NOISe:FIGure:MARKer<x>:X?

Returns <NRf> is the power difference value between MR and the selected marker.

Examples `CALCULATE:NOISE:FIGURE:MARKER1:DELTA:Y?` might return `18.7647705078`, indicating the difference between MR (reference marker) and the selected marker (M1) is 18.76 dB.

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

Places the selected marker at the maximum value on the peak of the trace.

Conditions Measurement view: Noise Figure

The parameter <x> = 0 to 4

Group Calculate commands

Syntax `CALCulate:NOISe:FIGure:MARKer<x>:MAXimum`

Related Commands CALCulate:NOISe:FIGure:MARKer<x>:PEAK:HIGHer

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LOWer

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LEFT

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:RIGHt

Arguments None

Returns None

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:HIGHer (No Query Form)

Places the selected marker at the next highest peak on the trace.

Conditions Measurement view: Noise Figure

The parameter <x> = 0 to 4

Group Calculate commands

Syntax `CALCulate:NOISe:FIGure:MARKer<x>:PEAK:HIGHer`

Related Commands CALCulate:NOISe:GAIN:MARKer<x>:MAXimum?

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LEFT

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LOWer

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:RIGHt

Arguments None

Returns None

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

Places the selected marker at the next peak on the trace to the left of the selected marker.

Conditions Measurement view: Noise Figure

The parameter <x> = 0 to 4

Group Calculate commands

Syntax `CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LEFT`

Related Commands CALCulate:NOISe:FIGure:MARKer<x>:PEAK:RIGHt

CALCulate:NOISe:FIGure:MARKer<x>:MAXimum

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:HIGHer

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LOWer

Arguments None

Returns None

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LOWer (No Query Form)

Places the selected marker at the next lowest peak on the trace.

Conditions Measurement view: Noise Figure

The parameter <x> = 0 to 4

Group Calculate commands

Syntax `CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LOWer`

Related Commands CALCulate:NOISe:FIGure:MARKer<x>:MAXimum

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:HIGHer

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LEFT

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:RIGHt

Arguments None

Returns None

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

This command places the specified marker on a specified trace. The query form returns the trace on which the marker resides.

Conditions Measurement view: Noise Figure

The parameter <x> = 0 to 4

Group Calculate commands

Syntax `CALCulate:NOISe:FIGure:MARKer<x>:PEAK:RIGHt`

Related Commands CALCulate:NOISe:FIGure:MARKer<x>:MAXimum

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:HIGHer

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LEFT

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LOWer

Arguments None

Returns None

CALCulate:NOISe:FIGure:MARKer<x>:TRACe

Sets or queries which trace the specified marker is attached to.

Conditions Measurement view: Noise Figure

The parameter <x> = 0 to 4

Group Calculate commands

Syntax

```
CALCulate:NOISe:FIGure:MARKer<x>:TRACe {TRACE1|TRACE2|TRACE3}
CALCulate:NOISe:FIGure:MARKer<x>:TRACe?
```

Related Commands

CALCulate:NOISe:FIGure:MARKer<x>:MAXimum

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:HIGHer

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LEFT

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:LOWer

CALCulate:NOISe:FIGure:MARKer<x>:PEAK:RIGHt

Arguments

TRACE1 moves the specified marker to trace 1.

TRACE2 moves the specified marker to trace 2.

TRACE3 moves the specified marker to trace 3.

Returns

TRACE1 means the specified marker is on trace 1.

TRACE2 means the specified marker is on trace 2.

TRACE3 means the specified marker is on trace 3.

Examples

`CALCULATE:NOISE:FIGURE:MARKER1:TRACE?` might return `TRACE2`, indicating that M1 (marker 1) is on trace 2.

CALCulate:NOISe:FIGure:MARKer<x>:X? (Query Only)

Queries the frequency of the specified marker.

Conditions

Measurement view: Noise Figure

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

```
CALCulate:NOISe:FIGure:MARKer<x>:X?
```

Related Commands

CALCulate:NOISe:FIGure:MARKer<x>:Y?

Returns

<NR3> = a floating point value with an exponent. This is the frequency of the specified marker.

Examples `CALCULATE:NOISE:FIGURE:MARKER1:X?` might return `608.0000000000E+6`, indicating the frequency of M1 (marker 1) is 608 MHz.

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

Queries the vertical position of the specified marker.

Conditions Measurement view: Noise Figure

The parameter <x> = 0 to 4

Group Calculate commands

Syntax `CALCulate:NOISe:FIGure:MARKer<x>:Y?`

Related Commands CALCulate:NOISe:FIGure:MARKer<x>:X?

Returns <NR2> = a floating point value without an exponent. This is the vertical position (dB) of the specified marker.

Examples `CALCULATE:NOISE:FIGURE:MARKER1:Y?` might return `18.7647705078`, indicating the value of M1 (marker 1) is 18.76 dB.

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

Query returns the frequency difference between MR (reference marker) and the selected marker.

Conditions Measurement view: Gain

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid.

Group Calculate commands

Syntax `CALCulate:NOISe:GAIN:MARKer<x>:DELTa:X?`

Related Commands CALCulate:NOISe:GAIN:MARKer<x>:DELTa:Y?

Returns <NR2> = a floating point value without an exponent. This is the frequency difference value between MR and the selected marker.

Examples `CALCULATE:NOISE:GAIN:MARKER1:DELTA:X?` might return `100.0000`, indicating the frequency difference between MR (reference marker) and the selected marker (M1) is 100 dB.

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

Query returns the power difference value between MR (reference marker) and the selected marker.

Conditions Measurement view: Gain

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid.

Group Calculate commands

Syntax `CALCulate:NOISe:GAIN:MARKer<x>:DELTa:Y?`

Related Commands CALCulate:NOISe:GAIN:MARKer<x>:DELTa:X?

Returns <NR2> = a floating point value without an exponent. This is the power difference value between MR and the selected marker.

Examples `CALCULATE:NOISE:GAIN:MARKER1:DELTA:Y?` might return `18.7647705078`, indicating the difference between MR (reference marker) and the selected marker (M1) is 18.76 dB.

CALCulate:NOISe:GAIN:MARKer<x>:MAXimum? (Query Only)

Places the selected marker at the maximum value on the peak of the trace.

Conditions Measurement view: Gain

The parameter <x> = 0 to 4

Group Calculate commands

Syntax `CALCulate:NOISe:GAIN:MARKer<x>:MAXimum?`

Related Commands CALCulate:NOISe:GAIN:MARKer<x>:PEAK:HIGHer?

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LEFT?

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:RIGHt?

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LOWer?

Arguments None

Returns None

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:HIGHer? (Query Only)

Places the selected marker at the next highest peak on the trace.

Conditions Measurement view: Gain

The parameter <x> = 0 to 4

Group Calculate commands

Syntax `CALCulate:NOISe:GAIN:MARKer<x>:PEAK:HIGHer?`

Related Commands CALCulate:NOISe:GAIN:MARKer<x>:MAXimum?

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LEFT?

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:RIGHt?

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LOWer?

Arguments None

Returns None

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LEFT? (Query Only)

Places the selected marker at the next peak on the trace to the left of the selected marker.

Conditions Measurement view: Gain

The parameter <x> = 0 to 4

Group Calculate commands

Syntax `CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LEFT?`

Related Commands CALCulate:NOISe:GAIN:MARKer<x>:MAXimum?

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:HIGHer?

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:RIGHt?

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LOWer?

Arguments None

Returns None

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LOWer? (Query Only)

Places the selected marker at the next lowest peak on the trace.

Conditions Measurement view: Gain

The parameter <x> = 0 to 4

Group Calculate commands

Syntax `CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LOWer?`

Related Commands CALCulate:NOISe:GAIN:MARKer<x>:MAXimum?

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:HIGHer?

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:RIGHt?

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:HIGHer?

Arguments None

Returns None

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:RIGHt? (Query Only)

Places the specified marker on a specified trace. The query form returns the trace on which the marker resides.

Conditions Measurement view: Gain

The parameter <x> = 0 to 4

Group Calculate commands

Syntax `CALCulate:NOISe:GAIN:MARKer<x>:PEAK:RIGHt?`

Related Commands CALCulate:NOISe:GAIN:MARKer<x>:MAXimum?

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:HIGHer?

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:LEFT?

CALCulate:NOISe:GAIN:MARKer<x>:PEAK:HIGHer?

Arguments None

Returns None

CALCulate:NOISe:GAIN:MARKer<x>:TRACe

Sets or queries which trace the specified marker is attached to.

Conditions Measurement view: Gain

The parameter <x> = 0 to 4

Group Calculate commands

Syntax
```
CALCulate:NOISe:GAIN:MARKer<x>:TRACe { TRACE1 | TRACE2 | TRACE3 }
CALCulate:NOISe:GAIN:MARKer<x>:TRACe?
```

Arguments TRACE1 moves the specified marker to trace 1.

TRACE2 moves the specified marker to trace 2.

TRACE3 moves the specified marker to trace 3.

Returns TRACE1 means the specified marker is on trace 1.

TRACE2 means the specified marker is on trace 2.

TRACE3 means the specified marker is on trace 3.

Examples `CALCULATE:NOISE:GAIN:MARKER1:TRACE?` might return `TRACE2`, indicating that M1 (marker 1) is on trace 2.

CALCulate:NOISe:GAIN:MARKer<x>:X? (Query Only)

This command queries the frequency of the specified marker.

Conditions Measurement view: Gain

The parameter <x> = 0 to 4

Group Calculate commands

Syntax
```
CALCulate:NOISe:GAIN:MARKer<x>:X?
```

Related Commands CALCulate:NOISe:GAIN:MARKer<x>:Y?

Returns <NR3> = a floating point value with an exponent. This is the frequency of the specified marker.

Examples `CALCULATE:NOISE:GAIN:MARKER1:X?` might return `608.0000000000E+6`, indicating the frequency of M1 (marker 1) is 608 MHz.

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

Queries the vertical position (dB) of the specified marker.

Conditions Measurement view: Gain

The parameter <x> = 0 to 4

Group Calculate commands

Syntax `CALCulate:NOISe:GAIN:MARKer<x>:Y?`

Related Commands CALCulate:NOISe:TEMPerature:MARKer<x>:DELTa:X?

Returns <NR2> = a floating point value without an exponent. This is the vertical position (dB) of the specified marker.

Examples `CALCULATE:NOISE:GAIN:MARKER1:Y?` might return `18.7647705078`, indicating the value of M1 (marker 1) is 18.76 dB.

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

Returns the frequency difference between MR (reference marker) and the selected marker.

Conditions Measurement view: Noise Temperature

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid.

Group Calculate commands

Syntax `CALCulate:NOISe:TEMPerature:MARKer<x>:DELTa:X?`

Related Commands CALCulate:NOISe:TEMPerature:MARKer<x>:DELTa:Y?

Returns <NR2> = a floating point value without an exponent. This is the frequency difference value between MR and the selected marker.

Examples `CALC:NOIS:TEMP:MARK1:DELT:X?` might return `100.0000`, indicating the frequency difference between MR (reference marker) and the selected marker (M1) is 100 dB.

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

Returns the temperature difference between MR (reference marker) and the selected marker.

Conditions Measurement view: Noise Temperature

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid.

Group Calculate commands

Syntax `CALCulate:NOISe:TEMPerature:MARKer<x>:DELTa:Y?`

Related Commands CALCulate:NOISe:TEMPerature:MARKer<x>:DELTa:X?

Returns <NR3> = a floating point value with an exponent. This is the temperature difference value between MR and the selected marker.

Examples `CALC:NOIS:TEMP:MARK1:DELT:Y?` might return `2.3663560181E+3`, indicating the difference between MR (reference marker) and the selected marker (M1) is 2.36 kK.

CALCulate:NOISe:TEMPerature:MARKer<x>:MAXimum? (Query Only)

Places the selected marker at the maximum value on the peak of the trace.

Conditions Measurement view: Noise Temperature

The parameter <x> = 0 to 4

Group Calculate commands

Syntax `CALCulate:NOISe:TEMPerature:MARKer<x>:MAXimum?`

Related Commands CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:HIGHer?

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LEFT?

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LOWer?

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:RIGHt?

Arguments None

Returns None

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:HIGHer? (Query Only)

Places the selected marker at the highest peak on the trace.

Conditions Measurement view: Noise Temperature

The parameter <x> = 0 to 4

Group Calculate commands

Syntax `CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:HIGHer?`

Related Commands CALCulate:NOISe:TEMPerature:MARKer<x>:MAXimum?

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LEFT?

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LOWer?

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:RIGHt?

Arguments None

Returns None

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LEFT? (Query Only)

Places the selected marker at the next peak on the trace to the left of the selected marker.

Conditions Measurement view: Noise Temperature

The parameter <x> = 0 to 4

Group Calculate commands

Syntax `CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LEFT?`

Related Commands CALCulate:NOISe:TEMPerature:MARKer<x>:MAXimum?

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:HIGHer?

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LOWer?

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:RIGHt?

Arguments None

Returns None

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LOWer? (Query Only)

Places the selected marker at the next lowest peak on the trace.

Conditions Measurement view: Noise Temperature

The parameter <x> = 0 to 4

Group Calculate commands

Syntax `CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LOWer?`

Related Commands CALCulate:NOISe:TEMPerature:MARKer<x>:MAXimum?

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:HIGHer?

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LEFT?

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:RIGHt?

Arguments None

Returns None

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:RIGHt? (Query Only)

This command places the specified marker on a specified trace. The query form returns the trace on which the marker resides.

Conditions Measurement view: Noise Temperature

The parameter <x> = 0 to 4

Group Calculate commands

Syntax `CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:RIGHt?`

Related Commands CALCulate:NOISe:TEMPerature:MARKer<x>:MAXimum?

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:HIGHer?

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LOWer?

CALCulate:NOISe:TEMPerature:MARKer<x>:PEAK:LEFT?

Arguments None

Returns None

CALCulate:NOISe:TEMPerature:MARKer<x>:TRACe

Sets or queries which trace the specified marker is attached to.

Conditions Measurement view: Noise Temperature

The parameter <x> = 0 to 4

Group Calculate commands

Syntax
```
CALCulate:NOISe:TEMPerature:MARKer<x>:TRACe { TRACE1 | TRACE2 | TRACE3 }
CALCulate:NOISe:TEMPerature:MARKer<x>:TRACe?
```

Arguments

TRACE1 moves the specified marker to trace 1.

TRACE2 moves the specified marker to trace 2.

TRACE3 moves the specified marker to trace 3.

Returns

TRACE1 means the specified marker is on trace 1.

TRACE2 means the specified marker is on trace 2.

TRACE3 means the specified marker is on trace 3.

Examples

`CALC:NOIS:TEMP:MARK1:TRAC?` might return `TRACE2`, indicating that M1 (marker 1) is on trace 2.

CALCulate:NOISe:TEMPerature:MARKer<x>:X? (Query Only)

Queries the frequency of the specified marker.

Conditions

Measurement view: Noise Temperature

The parameter <x> = 0 to 4

Group

Calculate commands

Syntax

`CALCulate:NOISe:TEMPerature:MARKer<x>:X?`

Related Commands

CALCulate:NOISe:TEMPerature:MARKer<x>:Y?

Returns

<NR2> = a floating point value without an exponent. This is the frequency of the specified marker.

Examples

`CALC:NOIS:TEMP:MARK1:X?` might return `608.0000000000E+6`, indicating the frequency of M1 (marker 1) is 608 MHz.

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

Queries the temperature (Kelvin) at the specified marker.

Conditions

Measurement view: Noise Temperature

The parameter <x> = 0 to 4

Group Calculate commands

Syntax `CALCulate:NOISe:TEMPerature:MARKer<x>:Y?`

Related Commands CALCulate:NOISe:TEMPerature:MARKer<x>:X?

Returns <NR3> = a floating point value with an exponent. This is the temperature at the specified marker.

Examples `CALC:NOIS:TEMP:MARK1:Y?` might return `1.4544410400E+3`, indicating the value at M1 (marker 1) is 1.45 kK.

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

Queries the frequency difference between MR (reference marker) and the selected marker.

Conditions Measurement view: Y Factor

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid.

Group Calculate commands

Syntax `CALCulate:NOISe:YFACtor:MARKer<x>:DELTa:X?`

Related Commands CALCulate:NOISe:YFACtor:MARKer<x>:DELTa:Y?

Returns <NR2> = a floating point value without an exponent. This is the frequency difference value between MR and the selected marker.

Examples `CALC:NOIS:YFAC:MARK1:DELT:X?` might return `100.0000`, indicating the frequency difference between MR (reference marker) and the selected marker (M1) is 100 dB.

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

Queries the power difference value between MR (reference marker) and the selected marker.

Conditions Measurement view: Y Factor

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid.

Group Calculate commands

Syntax `CALCulate:NOISe:YFACtor:MARKer<x>:DELTa:Y?`

Related Commands CALCulate:NOISe:YFACtor:MARKer<x>:DELTa:X?

Returns <NR2> = a floating point value without an exponent. This is the power difference value between MR and the selected marker.

Examples `CALC:NOIS:YFAC:MARK1:DELT:Y?` might return `18.7647705078`, indicating the difference between MR (reference marker) and the selected marker (M1) is 18.76 dB.

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

Places the selected marker at the maximum value on the peak of the trace.

Conditions Measurement view: Y Factor

The parameter <x> = 0 to 4

Group Calculate commands

Syntax `CALCulate:NOISe:YFACtor:MARKer<x>:MAXimum`

Related Commands CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:HIGHer?

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LEFT?

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LOWer?

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:RIGHt?

Arguments None

Returns None

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:HIGHer? (Query Only)

This command places the selected marker at the next highest peak on the trace.

Conditions Measurement view: Y Factor

The parameter <x> = 0 to 4

Group enter command group

Syntax `CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:HIGHer?`

Related Commands CALCulate:NOISe:YFACtor:MARKer<x>:MAXimum

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LEFT?

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LOWer?

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:RIGHt?

Arguments None

Returns None

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LEFT? (Query Only)

This command places the selected marker at the next peak on the trace to the left of the selected marker.

Conditions Measurement view: Y Factor

The parameter <x> = 0 to 4

Group Calculate commands

Syntax `CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LEFT?`

Related Commands CALCulate:NOISe:YFACtor:MARKer<x>:MAXimum

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:HIGHer?

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LOWer?

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:RIGHt?

Arguments None

Returns None

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LOWer? (Query Only)

Places the selected marker at the next lowest peak on the trace.

Conditions Measurement view: Y Factor

The parameter <x> = 0 to 4

Group Calculate commands

Syntax `CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LOWer?`

Related Commands CALCulate:NOISe:YFACtor:MARKer<x>:MAXimum

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LEFT?

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:HIGHer?

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:RIGHt?

Arguments None

Returns None

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:RIGHt? (Query Only)

Places the specified marker on a specified trace. The query form returns the trace on which the marker resides.

Conditions Measurement view: Y Factor

The parameter <x> = 0 to 4

Group Calculate commands

Syntax

```
CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:RIGHt?
```

Related Commands CALCulate:NOISe:YFACtor:MARKer<x>:MAXimum

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LEFT?

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:LOWer?

CALCulate:NOISe:YFACtor:MARKer<x>:PEAK:HIGHer?

Arguments None

Returns None

CALCulate:NOISe:YFACtor:MARKer<x>:TRACe

Sets or queries which trace the specified marker is attached to.

Conditions Measurement view: Y Factor

The parameter <x> = 0 to 4

Group Calculate commands

Syntax

```
CALCulate:NOISe:YFACtor:MARKer<x>:TRACe {TRACE1|TRACE2|TRACE3}
CALCulate:NOISe:YFACtor:MARKer<x>:TRACe?
```

Arguments TRACE1 moves the specified marker to trace 1.

TRACE2 moves the specified marker to trace 2.

TRACE3 moves the specified marker to trace 3.

Returns TRACE1 means the specified marker is on trace 1.

TRACE2 means the specified marker is on trace 2.

TRACE3 means the specified marker is on trace 3.

Examples `CALC:NOIS:YFAC:MARK1:TRAC?` might return `TRACE2`, indicating that M1 (marker 1) is on trace 2.

CALCulate:NOISe:YFACtor:MARKer<x>:X? (Query Only)

Queries the frequency of the specified marker.

Conditions Measurement view: Y Factor

The parameter <x> = 0 to 4

Group Calculate commands

Syntax `CALCulate:NOISe:YFACtor:MARKer<x>:X?`

Related Commands CALCulate:NOISe:YFACtor:MARKer<x>:Y?

Returns <NR3> = a floating point value with an exponent. This is the frequency of the specified marker.

Examples `CALC:NOIS:YFAC:MARK1:X?` might return `608.0000000000E+6`, indicating the frequency of M1 (marker 1) is 608 MHz.

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

This command queries the vertical position (dB) of the specified marker.

Conditions Measurement view: Y Factor

The parameter <x> = 0 to 4

Group Calculate commands

Syntax `CALCulate:NOISe:YFACtor:MARKer<x>:Y?`

Related Commands CALCulate:NOISe:YFACtor:MARKer<x>:X?

Returns <NR2> = a floating point value without an exponent. This is the dB of the specified marker.

Examples `CALCULATE:NOISE:YFACTOR:MARKER1:Y?` might return `18.7647705078`, indicating the value of M1 (marker 1) is 18.76 dB.

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

Returns the delta marker frequency for the selected marker in the Occupied Bandwidth measurement.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Occupied Bandwidth

Group Calculate commands

Syntax `CALCulate:OBWidth:MARKer<x>:DELTa:X?`

Related Commands CALCulate:OBWidth:MARKer<x>:DELTa:Y?

Arguments None

Returns `<NRf>` Delta marker frequency for the selected marker.

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

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

Returns the delta marker amplitude for the selected marker in the Occupied Bandwidth measurement.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Occupied Bandwidth

Group Calculate commands

Syntax `CALCulate:OBWidth:MARKer<x>:DELTa:Y?`

Related Commands CALCulate:OBWidth:MARKer<x>:DELTa:X?

Arguments None

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

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

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

Moves the selected marker to the highest peak on the trace in the Occupied Bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Calculate commands

Syntax `CALCulate:OBWidth:MARKer<x>:MAXimum`

Arguments None

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

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

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

Conditions Measurement views: Occupied Bandwidth

Group Calculate commands

Syntax `CALCulate:OBWidth:MARKer<x>:PEAK:HIGHer`

Related Commands CALCulate:OBWidth:MARKer<x>:PEAK:LOWer

Arguments None

Examples `CALCULATE:OBWIDTH:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

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

Moves the selected marker to the next peak to the left on the trace in the Occupied Bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Calculate commands

Syntax `CALCulate:OBWidth:MARKer<x>:PEAK:LEFT`

Related Commands CALCulate:OBWidth:MARKer<x>:PEAK:RIGHt

Arguments None

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

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

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

Conditions Measurement views: Occupied Bandwidth

Group Calculate commands

Syntax `CALCulate:OBWidth:MARKer<x>:PEAK:LOWer`

Related Commands CALCulate:OBWidth:MARKer<x>:PEAK:HIGHer

Arguments None

Examples `CALCULATE:OBWIDTH:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

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

Moves the selected marker to the next peak to the right on the trace in the Occupied Bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Calculate commands

Syntax `CALCulate:OBWidth:MARKer<x>:PEAK:RIGHt`

Related Commands CALCulate:OBWidth:MARKer<x>:PEAK:LEFT

Arguments None

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

CALCulate:OBWidth:MARKer<x>[:SET]:CENTer (No Query Form)

Sets the center frequency to the value at the marker position in the Occupied Bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Calculate commands

Syntax `CALCulate:OBWidth:MARKer<x>[:SET]:CENTer`

Arguments None

Examples `CALCULATE:OBWIDTH:MARKER1:SET:CENTER` sets the center frequency to the value at the Marker 1 position.

CALCulate:OBWidth:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the Occupied Bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Calculate commands

Syntax
```
CALCulate:OBWidth:MARKer<x>:X <value>
CALCulate:OBWidth:MARKer<x>:X?
```

Related Commands CALCulate:OBWidth:MARKer<x>:Y?

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

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

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

Queries the marker amplitude of the selected marker in the Occupied Bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Calculate commands

Syntax `CALCulate:OBWidth:MARKer<x>:Y?`

Related Commands CALCulate:OBWidth:MARKer<x>:X

Arguments None

Returns `<NRf>` Marker amplitude of the selected marker.

Examples `CALCULATE:OBWIDTH:MARKER1:Y?` might return -34.28 indicating Marker 1 (M1) is at -34.28 dBm.

CALCulate:OFDM:CONSte:MARKer<x>:FREQuency

Sets or queries the frequency position of the selected marker in the OFDM Constellation view.

Conditions Measurement views: OFDM

Group Calculate commands

Syntax
```
CALCulate:OFDM:CONSte:MARKer<x>:FREQuency <value>
CALCulate:OFDM:CONSte:MARKer<x>:FREQuency?
```

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

Returns <NRf> The value type depends on where the marker is placed.

Examples CALCulate:OFDM:CONSte:MARKer1:FREQuency? might return 800 MHz, indicating that Marker 1 M1 is positioned at the 800 MHz position in the OFDM Constellation view.

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

Queries the marker magnitude of the selected marker in the OFDM Constellation view.

Conditions Measurement views: OFDM

Group Calculate commands

Syntax CALCulate:OFDM:CONSte:MARKer<x>:MAGNitude?

Arguments None

Returns <NRf> The magnitude for the selected marker in dB.

Examples CALCulate:OFDM:CONSte:MARKer1:MAGNitude might return 35 indicating readout of Marker 1 (M1) is 35 dB.

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

Queries the phase readout of the selected marker in the OFDM Constellation measurement view.

Conditions Measurement views: OFDM

Group Calculate commands

Syntax CALCulate:OFDM:CONSte:MARKer<x>:PHASe?

Arguments None

Returns <NRf> The phase readout for the selected marker.

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

CALCulate:OFDM:CONSte:MARKer<x>:TIME

Sets or queries the X position of the selected marker in the OFDM Constellation measurement view.

Conditions Measurement views: OFDM

Group Calculate commands

Syntax
```
CALCulate:OFDM:CONSte:MARKer<x>:TIME <value>
CALCulate:OFDM:CONSte:MARKer<x>:TIME?
```

Arguments <value::=NRf> specifies the marker time.

Examples CALCulate:OFDM:CONSte:MARKer1:TIME –234.5us places Marker 1 (M1) at the –234.5 µs on the trace.

CALCulate:OFDM:CONSte:MARKer<x>:TYPE? (Query Only)

Queries the data type of the selected marker in OFDM Constellation view.

Conditions Measurement views: OFDM

Group Calculate commands

Syntax CALCulate:OFDM:CONSte:MARKer<x>:TYPE?

Arguments None

Returns `1.000` if the marker type is a data marker.

`0.000` if the marker type is a pilot marker.

Examples `CALCulate:OFDM:CONSte:MARKer1:TYPE?` might return `1.000` indicating Marker 1 (M1) is a data marker.

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

Queries the value of the specified marker in OFDM Constellation view.

Conditions Measurement views: OFDM

Group Calculate commands

Syntax `CALCulate:OFDM:CONSte:MARKer<x>:VALue?`

Arguments None

Returns `<NRf>` The value readout for the selected marker.

Examples `CALCulate:OFDM:CONSte:MARKer2:VALue?` might return 2.00, indicating the value readout of Marker 2 (M2) is 2.

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

Returns the delta marker frequency for the selected marker in the OFDM Channel response view.

Conditions Measurement views: OFDM

Group Calculate commands

Syntax `CALCulate:OFDM:CRESponse:MARKer<x>:DELTa:X?`

Arguments None

Returns <NRf> Delta Marker frequency for the selected marker.

Examples CALCulate:OFDM:CRESponse:MARKer2:DELTa:X might return 1.28E+6 indicating that the delta marker frequency for Marker 2 (M2) is 1.28 MHz.

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

Returns the delta marker frequency for the selected marker in the OFDM Channel response view.

Conditions Measurement views: OFDM

Group Calculate commands

Syntax CALCulate:OFDM:CRESponse:MARKer<x>:DELTa:Y?

Arguments None

Returns <NRf> Delta Marker frequency for the selected marker.

Examples CALCulate:OFDM:CRESponse:MARKer2:DELTa:Y might return 1.28E+6 indicating that the delta marker frequency for Marker 2 (M2) is 1.28 MHz.

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

Moves the selected marker to the highest peak in the OFDM Channel response view.

Conditions Measurement views: OFDM

Group Calculate commands

Syntax CALCulate:OFDM:CRESponse:MARKer<x>:MAXimum

Arguments None

Examples `CALCulate:OFDM:CRESponse:MARKer1:MAXimum` moves Marker 1 (M1) to the highest peak in the OFDM Channel response view.

CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude in the OFDM Channel response view.

Conditions Measurement views: OFDM

Group Calculate commands

Syntax `CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:HIGHer`

Arguments None

Examples `CALCulate:OFDM:CRESponse:MARKer1:PEAK:HIGHer` moves Marker 1 (M1) to the next peak higher in amplitude in the OFDM Channel response view.

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

Moves the selected marker to the next peak to the left in the OFDM Channel response view.

Conditions Measurement views: OFDM

Group Calculate commands

Syntax `CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:LEFT`

Arguments None

Examples `CALCulate:OFDM:CRESponse:MARKer1:PEAK:LEFT` moves Marker1 (M1) to the next peak to the left in the OFDM Channel response view.

CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude in the OFDM Channel response view.

Conditions Measurement views: OFDM

Group Calculate commands

Syntax `CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:LOWer`

Arguments None

Examples `CALCulate:OFDM:CRESponse:MARKer2:PEAK:LOWer` moves Marker 2 (M2) to the next peak lower in amplitude in the OFDM Channel response view.

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

Moves the selected marker to the next peak to the right in the OFDM Channel response view.

Conditions Measurement views: OFDM

Group Calculate commands

Syntax `CALCulate:OFDM:CRESponse:MARKer<x>:PEAK:RIGHt`

Arguments None

Examples `CALCulate:OFDM:CRESponse:MARKer1:PEAK:RIGHt` moves Marker1 (M1) to the next peak to the right in the OFDM Channel response view.

CALCulate:OFDM:CRESponse:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the OFDM Channel response view.

Conditions Measurement views: OFDM

Group Calculate commands

Syntax
```
CALCulate:OFDM:CRESponse:MARKer<x>:X <value>
CALCulate:OFDM:CRESponse:MARKer<x>:X?
```

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

Examples `CALCulate:OFDM:CRESponse:MARKer1:X 800MHZ` places Marker 1 (M1) at 800 MHz on the trace.

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

Returns the amplitude of the selected marker in the OFDM Channel response view.

Conditions Measurement views: OFDM

Group Calculate commands

Syntax
```
CALCulate:OFDM:CRESponse:MARKer<x>:Y?
```

Arguments None

Returns <NRf> Amplitude of the selected marker in dB.

Examples CALCulate:OFDM:CRESponse:MARKer1:Y might return `33.393856E-3` indicating the amplitude of Marker 1 (M1) is 0.033 dB.

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

Positions the specified marker at the maximum point on the trace in the OFDM Spectral Flatness display.

Conditions Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group Calculate commands

Syntax `CALCulate:OFDM:FLATness:MARKer<x>:MAXimum`

Arguments None

Examples `CALC:OFDM:FLAT:MARK2:MAX` positions the Marker 2 (M2) at the maximum point on the trace.

CALCulate:OFDM:FLATness:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the specified marker to the next peak higher in amplitude on the trace on the OFDM Spectral Flatness display.

Conditions Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group Calculate commands

Syntax `CALCulate:OFDM:FLATness:MARKer<x>:PEAK:HIGHer`

Arguments None

Examples `CALC:OFDM:FLAT:MARK2:PEAK:HIGHer` moves Marker 2 (M2) to the next peak higher in amplitude on the trace.

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

Moves the specified marker to the next peak to the left on the trace in the OFDM Spectral Flatness display.

Conditions Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group Calculate commands

Syntax `CALCulate:OFDM:FLATness:MARKer<x>:PEAK:LEFT`

Arguments None

Examples `CALC:OFDM:FLAT:MARK2:PEAK:LEFT` moves Marker 2 (M2) to the next peak to the left on the trace .

CALCulate:OFDM:FLATness:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the marker to the next peak lower in amplitude on the trace on the OFDM Spectral Flatness display.

Conditions Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group Calculate commands

Syntax `CALCulate:OFDM:FLATness:MARKer<x>:PEAK:LOWer`

Arguments None

Examples `CALC:OFDM:FLAT:MARK2:PEAK:LOW` moves Marker 2 (M2) to the next peak lower on the trace.

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

Moves the marker to the next peak to the right on the trace on the OFDM Spectral Flatness display.

Conditions Measurement view: OFDM

This command requires Option 22, "OFDM Measurements".

Group Calculate commands

Syntax `CALCulate:OFDM:FLATness:MARKer<x>:PEAK:RIGHt`

Arguments None

Examples `CALC:OFDM:FLAT:MARK2:PEAK:RIGH` moves Marker 2 (M2) to the next peak to the next peak on the right on the trace.

CALCulate:OFDM:FLATness:MARKer<x>:X

Sets or queries the horizontal value at the selected marker position on the points trace in the OFDM Spectral Flatness display. The units are in Frequency or Subcarrier.

To set the units, use the command `[SENSe]:OFDM:UNIT:FREQuency`.

Conditions Measurement view: OFDM

This command requires Option 22, "OFDM Measurements".

Group Calculate commands

Syntax
```
CALCulate:OFDM:FLATness:MARKer<x>:X <NRf>
CALCulate:OFDM:FLATness:MARKer<x>:X?
```

Arguments Floating point number that specifies the horizontal position of the marker. Range: Start to Stop frequency (left to right edge of the horizontal axis). Using an out-of-range value causes an execution error (-222, "Data out of range").

Examples `CALC:OFDM:FLAT:MARK3:X 22` places Marker 3 at 22 Subcarrier on the trace (assuming the units have been set to Subcarrier).

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

Returns the value of the amplitude (vertical position) at the selected marker position on the OFDM Spectral Flatness display.

Conditions Measurement view: OFDM

This command requires Option 22, "OFDM Measurements".

Group Calculate commands

Syntax `CALCulate:OFDM:FLATness:MARKer<x>:Y?`

Returns Floating point number that indicates the amplitude at the selected marker position in dB.

Examples `CALC:OFDM:FLAT:MARK1:Y?` might return `33.393856E-3` indicating the amplitude at Marker 1 is 0.033 dB.

CALCulate:OFDM:TABLe:MARKer<x>:FREQuency

Sets or queries the frequency position of the selected marker in the OFDM Symbol table view.

Conditions Measurement views: OFDM

Group Calculate commands

Syntax
```
CALCulate:OFDM:TABLe:MARKer<x>:FREQuency <value>
CALCulate:OFDM:TABLe:MARKer<x>:FREQuency?
```

Arguments `<value>::=<NRf>` specifies the marker frequency.

Examples CALCulate:OFDM:TABLe:MARKer1:FREQuency 800MHZ places Marker 1 (M1) at 800 MHz in the symbol table.

CALCulate:OFDM:TABLe:MARKer<x>:TIMe

Sets or queries the marker time in the OFDM Symbol table view.

Conditions Measurement views: OFDM

Group Calculate commands

Syntax

```
CALCulate:OFDM:TABLe:MARKer<x>:TIMe <value>
CALCulate:OFDM:TABLe:MARKer<x>:TIMe?
```

Arguments <value>::=<NRf> specifies the marker time.

Examples `CALCulate:OFDM:TABLe:MARKer1:TIMe –234.5us` places Marker 1 (M1) at –234.5 µs in the symbol table.

CALCulate:OFDM:TABLe:MARKer<x>:VALue? (Query Only)

Queries the value readout of the selected marker in the OFDM Symbol table view.

Conditions Measurement views: OFDM

Group Calculate commands

Syntax `CALCulate:OFDM:TABLe:MARKer<x>:VALue?`

Arguments None

Returns `<NRf>` The value readout of the selected marker.

Examples `CALCulate:OFDM:TABLe:MARKer2:VALue?` might return 2.00 indicating the value readout of Marker 2 (M2) is 2.

CALCulate:P25:CONSte:MARKer<x>:DELTa:X[:TIME]? (Query Only)

Returns the delta marker time for the selected marker on the P25 Constellation display.

Conditions Measurement view: P25 Constellation

The parameter <x>=1 to 4; MARKer0 (reference marker) is invalid.

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group Calculate commands

Syntax `CALCulate:P25:CONSte:MARKer<x>:DELTa:X[:TIME]?`

Returns <NR2> = a floating point value without an exponent. This is the time for the selected Delta marker.

NOTE. *Use the `[SENSe]:P25:TIMe:UNITs` command to select the time unit: symbols (default) or seconds.*

Examples CALCulate:P25:CONSte:MARKer1:DELTa:X:TIME? might return 62.75, indicating that the Delta marker time is 62.75 symbols.

CALCulate:P25:CONSte:MARKer<x>:FDEViation? (Query Only)

Queries the frequency deviation of the selected marker in the P25 Constellation display.

Conditions Measurement view: P25 Constellation

The parameter <x> =0 to 4.

The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group Calculate commands

Syntax CALCulate:P25:CONSte:MARKer<x>:FDEViation?

Returns <NR2> = a floating point value without an exponent. This is the frequency deviation readout for the selected marker.

Examples CALC:P25:CONS:MARr1:FDEV? might return 102.82380000E+3, indicating the frequency deviation readout of Marker 1 (M1) is 102.824 kHz.

CALCulate:P25:CONSte:MARKer<x>:MAGNitude? (Query Only)

Queries the frequency deviation (for Freq Dev trace types) or magnitude (for IQ trace types) of the selected marker in the P25 Constellation display.

Conditions Measurement view: P25 Constellation

Group Calculate commands

Syntax CALCulate:P25:CONSte:MARKer<x>:MAGNitude?

Returns <NRf> is the frequency deviation or magnitude of the selected marker.

Examples

CALC:P25:CONS:MARK1:MAGN? might return 1.6257750459E+3 if the trace type is Freq Dev, indicating that Marker 1 (M1) has a frequency deviation of 1.626 kHz.

CALC:P25:CONS:MARK1:MAGN? might return 1.0048475788 if the trace type is IQ, indicating that Marker 1 (M1) has a magnitude of 1.005.

CALCulate:P25:CONSte:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the trace in the P25 Constellation display.

Conditions

Measurement view: P25 Constellation

The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group

Calculate commands

Syntax

CALCulate:P25:CONSte:MARKer<x>:MAXimum

Related Commands

CALCulate:P25:CONSte:MARKer<x>:PEAK:LEFT

CALCulate:P25:CONSte:MARKer<x>:PEAK:RIGHt

Examples

CALC:P25:CONS:MARK1:MAX positions Marker 1 (M1) at the symbol in the center of the time record.

CALCulate:P25:CONSte:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the trace in the P25 Constellation display.

Conditions

Measurement view: P25 Constellation

Group

Calculate commands

Syntax

CALCulate:P25:CONSte:MARKer<x>:PEAK:LEFT

Related Commands CALCulate:P25:CONSte:MARKer<x>:PEAK:RIGHt

CALCulate:P25:CONSte:MARKer<x>:MAXimum

Examples `CALC:P25:CONS:MARK1:PEAK:LEFT` moves Marker 1 (M1) in the time domain to the next lower symbol number.

CALCulate:P25:CONSte:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the trace in the P25 Constellation display.

Conditions Measurement view: P25 Constellation

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group Calculate commands

Syntax `CALCulate:P25:CONSte:MARKer<x>:PEAK:RIGHt`

Related Commands CALCulate:P25:CONSte:MARKer<x>:MAXimum

CALCulate:P25:CONSte:MARKer<x>:PEAK:LEFT

Examples `CALC:P25:CONS:MARK1:PEAK:RIGH` moves the Marker 1 (M1) in the time domain to the next higher symbol number to the right.

CALCulate:P25:CONSte:MARKer<x>:PHASe? (Query Only)

Queries the phase of the selected marker in the P25 Constellation display.

Conditions Measurement view: P25 Constellation

IQ trace type required.

Group Calculate commands

Syntax `CALCulate:P25:CONSte:MARKer<x>:PHASe?`

Returns <NR2> = a floating point value without an exponent. This is the phase value for the selected marker.

Examples `CALC:P25:CONS:MARK1:PHAS?` might return 41.4721108251, indicating the phase of Marker 1 (M1) is 41.5°.

CALCulate:P25:CONSte:MARKer<x>:SYMBol? (Query Only)

Queries the readout symbol of the selected marker in the P25 Constellation display.

Conditions Measurement view: P25 Constellation

The parameter <x>= 1 to 4; MARKer0 (reference marker) is invalid.

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group Calculate commands

Syntax `CALCulate:P25:CONSte:MARKer<x>:SYMBol?`

Returns <NR2> = a floating point value is the readout symbol for the selected marker.

Examples `CALCulate:P25:CONSte:MARKer1:SYMBol?` might return 62.0000000000, indicating the readout symbol of Marker 1 (M1) is 62.

CALCulate:P25:CONSte:MARKer<x>:Trace

Places the selected marker on the Demodulated I and Q versus Time trace. The query returns the name of the trace on which the marker resides.

Conditions Measurement view: P25 Constellation

The parameter <x>= 0 to 4.

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group Calculate commands

Syntax

```
CALCulate:P25:CONSte:MARKer<x>:Trace {Trace1 | Trace2}
CALCulate:P25:CONSte:MARKer<x>:Trace?
```

Arguments

Trace 1: sets the *I* trace marker on.

Trace 2: sets the *Q* trace marker on.

Returns

Trace 1 = *I* trace

Trace 2 = *Q* trace

Examples

`CALC:P25:CONS:MARK1:TRAC Trace1` places Marker 1 (M1) on the *I* trace.

CALCulate:P25:CONSte:MARKer<x>:VALue? (Query Only)

Queries the readout value of the specified marker in P25 Constellation display.

Conditions

Measurement view: P25 Constellation

Group

Calculate commands

Syntax

```
CALCulate:P25:CONSte:MARKer<x>:VALue?
```

Returns

<NRf> is the readout value for the selected marker.

Examples

`CALC:P25:CONS:MARK2:VAL?` might return 2.00, indicating the readout value of Marker 2 (M2) is 2.

CALCulate:P25:CONSte:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the P25 Constellation display.

Conditions

Measurement view: P25 Constellation

Group

Calculate commands

Syntax

```
CALCulate:P25:CONSte:MARKer<x>:X <value>
CALCulate:P25:CONSte:MARKer<x>:X?
```

Arguments

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

Returns

Offset set value(-500Hz t0 +500Hz).

Examples

`CALCulate:P25:PVTime:MARKer1:X 800MHz` places Marker 1 (M1) at 800 MHz on the trace. When used as a query, the return will show the horizontal position of the trace. For example, 800 MHz.

CALCulate:P25:EDIagram:MARKer<x>:DELTa:X[:TIME]? (Query Only)

Returns the delta marker time for the selected marker on the P25 Eye Diagram trace.

Conditions

Measurement view: P25 Eye Diagram

The parameter <x>=0 to 4. MARKer0 (reference marker) is invalid.

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group

Calculate commands

Syntax

```
CALCulate:P25:EDIagram:MARKer<x>:DELTa:X[:TIME]?
```

Related Commands

CALCulate:P25:EDIagram:MARKer<x>:DELTa:Y?

Returns

<NRf> is the Delta marker time for the selected marker. Use the `[SENSe]:P25:TIMe:UNITs` command to select the time unit: symbols (default) or seconds.

Examples

`CALCulate:P25:EDIagram:MARKer1:DELTa:X:TIME?` might return 62.750000000, Indicating that the delta marker time is 62.75 symbols.

CALCulate:P25:EDIagram:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker amplitude for the selected marker on the P25 Eye Diagram trace.

Conditions Measurement view: P25 Eye Diagram

The parameter <x>=1 to 4. MARKer0 (reference marker) is invalid.

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group Calculate commands

Syntax `CALCulate:P25:EDIagram:MARKer<x>:DELTa:Y?`

Related Commands CALCulate:P25:EDIagram:MARKer<x>:DELTa:X[:TIME]?

Returns <NRf> is the Delta marker amplitude for the selected marker.

Examples `CALCulate:P25:EDIagram:MARKer1:DELTa:Y?` might return -1.04300000, indicating that the delta marker amplitude is -1.043.

CALCulate:P25:EDIagram:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the P25 Eye Diagram trace.

Conditions Measurement view: P25 Eye Diagram

The parameter <x>=0 to 4.

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group Calculate commands

Syntax `CALCulate:P25:EDIagram:MARKer<x>:MAXimum`

Related Commands CALCulate:P25:EDIagram:MARKer<x>:PEAK:HIGHer

CALCulate:P25:EDIagram:MARKer<x>:PEAK:LEFT

CALCulate:P25:EDIagram:MARKer<x>:PEAK:LOWer

CALCulate:P25:EDIagram:MARKer<x>:PEAK:RIGHt

Arguments None

Returns None

Examples `CALCulate:P25:EDIagram:MARKer1:MAXImum` moves Marker 1 (M1) to the highest peak on the trace.

CALCulate:P25:EDIagram:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on the P25 Eye Diagram trace.

Conditions Measurement view: P25 Eye Diagram

The parameter <x>=0 to 4.

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group Calculate commands

Syntax `CALCulate:P25:EDIagram:MARKer<x>:PEAK:HIGHer`

Related Commands CALCulate:P25:EDIagram:MARKer<x>:MAXimum

CALCulate:P25:EDIagram:MARKer<x>:PEAK:LEFT

CALCulate:P25:EDIagram:MARKer<x>:PEAK:LOWer

CALCulate:P25:EDIagram:MARKer<x>:PEAK:RIGHt

Arguments None

Returns None

Examples `CALCulate:P25:EDIagram:MARKer1:PEAK:HIGHer` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

CALCulate:P25:EDIagram:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the P25 Eye Diagram trace.

Conditions Measurement view: P25 Eye Diagram

The parameter <x>=0 to 4.

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group Calculate commands

Syntax `CALCulate:P25:EDIagram:MARKer<x>:PEAK:LEFT`

Related Commands CALCulate:P25:EDIagram:MARKer<x>:MAXimum

CALCulate:P25:EDIagram:MARKer<x>:PEAK:HIGHer

CALCulate:P25:EDIagram:MARKer<x>:PEAK:LOWer

CALCulate:P25:EDIagram:MARKer<x>:PEAK:RIGHt

Arguments None

Returns None

Examples `CALCulate:P25:EDIagram:MARKer1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:P25:EDIagram:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the P25 Eye Diagram trace.

Conditions Measurement view: P25 Eye Diagram

The parameter <x>=0 to 4.

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group Calculate commands

Syntax `CALCulate:P25:EDIagram:MARKer<x>:PEAK:LOWer`

Related Commands CALCulate:P25:EDIagram:MARKer<x>:MAXimum

CALCulate:P25:EDIagram:MARKer<x>:PEAK:LEFT

CALCulate:P25:EDIagram:MARKer<x>:PEAK:HIGHer

CALCulate:P25:EDIagram:MARKer<x>:PEAK:RIGHt

Arguments None

Returns None

Examples `CALCulate:P25:EDIagram:MARKer1:PEAK:LOWer` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:P25:EDIagram:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the P25 Eye Diagram trace.

Conditions Measurement view: P25 Eye Diagram

The parameter <x>=0 to 4.

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group Calculate commands

Syntax `CALCulate:P25:EDIagram:MARKer<x>:PEAK:RIGHt`

Related Commands CALCulate:P25:EDIagram:MARKer<x>:MAXimum

CALCulate:P25:EDIagram:MARKer<x>:PEAK:LEFT

CALCulate:P25:EDIagram:MARKer<x>:PEAK:HIGHer

CALCulate:P25:EDIagram:MARKer<x>:PEAK:LOWer

Arguments None

Returns None

Examples `CALCulate:P25:EDIagram:MARKer1:PEAK:RIGHt` moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:P25:EDIagram:MARKer<x>:TRACe

Sets the selected marker on the I or Q trace, or queries the trace that the specified marker is on, in the P25 Eye Diagram display.

Conditions Measurement view: P25 Eye Diagram

The parameter <x>=0 to 4.

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group Calculate commands

Syntax

```
CALCulate:P25:EDIagram:MARKer<x>:TRACe { TRACe1 | TRACe2 }
CALCulate:P25:EDIagram:MARKer<x>:TRACe?
```

Arguments TRACe1 places the specified marker on the *I* trace.

TRACe2 places the specified marker on the *Q* trace.

Returns TRACe1 means the specified marker is on the *I* trace.

TRACe2 means the specified marker is on the *Q* trace.

Examples `CALCulate:P25:EDIagram:MARKer1:TRACe TRACe1` places Marker 1 (M1) on the I trace in the P25 Eye Diagram display.

CALCulate:P25:EDIagram:MARKer<x>:X

Sets or queries the horizontal position (time) of the selected marker in the P25 Eye Diagram display.

Conditions Measurement view: P25 Eye Diagram

The parameter <x>=0 to 4.

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group Calculate commands

Syntax
```
CALCulate:P25:EDIagram:MARKer<x>:X <NRf>
CALCulate:P25:EDIagram:MARKer<x>:X?
```

Related Commands CALCulate:P25:EDIagram:MARKer<x>:Y

Arguments <NRf> specifies the horizontal position (time) of the marker.

NOTE. *Use the `[SENSe]:P25:TIMe:UNITs` command to select the time unit: Symbols (default) or seconds.*

Returns <NRf> is the horizontal position of the selected marker.

Examples `CALC:P25:EDI:MARK1:X 38.5` places Marker 1 (M1) at 38.5 symbols on the trace.

CALCulate:P25:EDIagram:MARKer<x>:Y

Sets or queries the vertical position of the selected marker in the P25 Eye Diagram display.

Conditions Measurement view: P25 Eye Diagram

The parameter <x>=0 to 4.

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group Calculate commands

Syntax
```
CALCulate:P25:EDIagram:MARKer<x>:Y <value>
CALCulate:P25:EDIagram:MARKer<x>:Y?
```

Related Commands CALCulate:P25:EDIagram:MARKer<x>:X

Arguments <value>:: =<NRf> specifies the vertical position of the marker.

Returns <NRf> is the vertical position of the selected marker.

Examples `CALC:P25:EDI:MARK1:Y?` might return 571.8E-3, indicating Marker 1 (M1) is at 0.5718.

CALCulate:P25:PVTime:MARKer<x>:DELTa:X[:TIME]? (Query Only)

Returns the delta marker time (seconds) for the selected marker on the Power vs. Time display.

Conditions Measurement view: P25 Power vs. Time

The parameter <x>= 1 to 4; MARKer0 (reference marker) is invalid.

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group Calculate commands

Syntax `CALCulate:P25:PVTime:MARKer<x>:DELTa:X[:TIME]?`

Related Commands CALCulate:P25:PVTime:MARKer<x>:DELTa:Y[:TIME]?

SENSe:P25:TIMe:UNITs

Returns <NRf> is the Delta marker time for the selected marker.

NOTE. *Use the `SENSe:P25:TIME:UNITs` command to select the time unit symbols (default) or seconds.*

Examples `CALC:P25:PVT:MARK1:DELT:X?` might return -3.99372800000E-3, indicating that the delta marker 1 (M1) time is -3.937 ms.

CALCulate:P25:PVTime:MARKer<x>:DELTa:Y[:TIME]? (Query Only)

Returns the Delta marker amplitude (dB) for the selected marker on the Power vs. Time display.

Conditions Measurement view: P25 Power vs. Time

The parameter <x>= 1 to 4; MARKer0 (reference marker) is invalid.

The specified marker must be activated using the `CALCulate:MARKer:ADD` command.

Group Calculate commands

Syntax `CALCulate:P25:PVTime:MARKer<x>:DELTa:Y[:TIME]?`

Related Commands CALCulate:P25:PVTime:MARKer<x>:DELTa:X[:TIME]?

Returns <NR3> is a floating point value with an exponent. This is the Delta marker amplitude.

Examples `CALC:P25:PVT:MARK1:DELT:Y?` might return 552.6123046875E-3, indicating that the Delta marker 1 (M1) amplitude is 0.55 dB.

CALCulate:P25:PVTime:MARKer<x>:Maximum (No Query Form)

Moves the selected marker to the highest peak on the trace in the Power vs. Time display.

Conditions Measurement view: P25 Power vs. Time

Group Calculate commands

Syntax `CALCulate:P25:PVTime:MARKer<x>:Maximum`

Related Commands CALCulate:P25:PVTime:MARKer<x>:PEAK:LOWer

CALCulate:P25:PVTime:MARKer<x>:PEAK:HIGHer

CALCulate:P25:PVTime:MARKer<x>:PEAK:LEFT

CALCulate:P25:PVTime:MARKer<x>:PEAK:RIGHt

Arguments None

Returns None

Examples `CALC:P25:PVT:MARK1:PEAK:MAX` moves Marker 1 (M1) to the highest peak on the trace.

CALCulate:P25:PVTime:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on the trace in the Power vs. Time display.

Conditions Measurement view: P25 Power vs. Time

Group Calculate commands

Syntax `CALCulate:P25:PVTime:MARKer<x>:PEAK:HIGHer`

Related Commands CALCulate:P25:PVTime:MARKer<x>:PEAK:LOWer

CALCulate:P25:PVTime:MARKer<x>:Maximum

CALCulate:P25:PVTime:MARKer<x>:PEAK:LEFT

CALCulate:P25:PVTime:MARKer<x>:PEAK:RIGHt

Arguments None

Returns None

Examples `CALC:P25:PVT:MARK1:PEAK:HIGH` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

CALCulate:P25:PVTime:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the trace in the Power vs. Time display.

Conditions Measurement view: P25 Power vs. Time

Group Calculate commands

Syntax `CALCulate:P25:PVTime:MARKer<x>:PEAK:LEFT`

Related Commands CALCulate:P25:PVTime:MARKer<x>:PEAK:LOWer

CALCulate:P25:PVTime:MARKer<x>:Maximum

CALCulate:P25:PVTime:MARKer<x>:PEAK:HIGHer

CALCulate:P25:PVTime:MARKer<x>:PEAK:RIGHt

Arguments None

Returns None

Examples `CALC:P25:PVT:MARK1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:P25:PVTime:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the trace in the Power vs. Time display.

Conditions Measurement view: P25 Power vs. Time

Group Calculate commands

Syntax `CALCulate:P25:PVTime:MARKer<x>:PEAK:LOWer`

Related Commands CALCulate:P25:PVTime:MARKer<x>:PEAK:HIGHer

CALCulate:P25:PVTime:MARKer<x>:Maximum

CALCulate:P25:PVTime:MARKer<x>:PEAK:LEFT

CALCulate:P25:PVTime:MARKer<x>:PEAK:RIGHt

Arguments None

Returns None

Examples `CALC:P25:PVT:MARK1:PEAK:LOW` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:P25:PVTime:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the trace in the Power vs. Time display.

Conditions Measurement view: Power vs. Time

Group Calculate commands

Syntax `CALCulate:P25:PVTime:MARKer<x>:PEAK:RIGHt`

Related Commands CALCulate:P25:PVTime:MARKer<x>:PEAK:HIGHer

CALCulate:P25:PVTime:MARKer<x>:Maximum

CALCulate:P25:PVTime:MARKer<x>:PEAK:LEFT

CALCulate:P25:PVTime:MARKer<x>:PEAK:LOWer

Arguments None

Returns None

Examples `CALC:P25:PVT:MARK1:PEAK:RIGH` moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:P25:PVTime:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the Power vs. Time display.

Conditions Measurement view: P25 Power vs. Time

Group Calculate commands

Syntax
```
CALCulate:P25:PVTime:MARKer<x>:X <value>
CALCulate:P25:PVTime:MARKer<x>:X?
```

Related Commands CALCulate:P25:PVTime:MARKer<x>:Y

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

Returns <value>::=<NRf> specifies the horizontal position of the marker.

Examples `CALC:P25:PVT:MARK1:X 800MHz` places Marker 1 (M1) at 800 MHz on the trace. When used as a query, the return will show the horizontal position of the trace. For example, 800 MHz.

CALCulate:P25:PVTime:MARKer<x>:Y

Sets or queries the vertical position of the selected marker in the P25 Power vs. Time display.

Conditions Measurement view: P25 Power vs. Time

Group Calculate commands

Syntax
```
CALCulate:P25:PVTime:MARKer<x>:Y <value>
CALCulate:P25:PVTime:MARKer<x>:Y?
```

Related Commands CALCulate:P25:PVTime:MARKer<x>:X

Arguments <value>::=<NRf> specifies the vertical position of the marker.

Returns <NR2> = floating point value without an exponent.

Examples `CALC:P25:PVT:MARK1:Y?` might return -67.1001052856, indicating that the horizontal position of Marker 1 (M1) is -67.10 dBm.

`CALC:P25:PVT:MARK1:Y -75.66dBm` places Marker 1 (M1) at -75.66 on the trace.

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Phase error versus Time

Group Calculate commands

Syntax `CALCulate:PERRor:MARKer<x>:DELTa:X?`

Related Commands CALCulate:PERRor:MARKer<x>:DELTa:Y?

Arguments None

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

The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples `CALCULATE:PERROR:MARKER1:DELTA:X?` might return `9.52`, indicating that the delta marker time is 9.52 symbols.

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

Returns the delta marker phase for the selected marker in the Phase error versus Time measurement.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Phase error versus Time

Group Calculate commands

Syntax `CALCulate:PERRor:MARKer<x>:DELTa:Y?`

Related Commands CALCulate:PERRor:MARKer<x>:DELTa:X?

Arguments None

Returns `<NRf>` Delta marker phase for the selected marker in degrees.

Examples `CALCULATE:PERROR:MARKER1:DELTA:Y?` might return `-7.93`, indicating that the delta marker phase is -7.93 °.

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

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

Conditions Measurement views: Phase error versus Time

Group Calculate commands

Syntax `CALCulate:PERRor:MARKer<x>:MAXimum`

Arguments None

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

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

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

Conditions Measurement views: Phase error versus Time

Group Calculate commands

Syntax `CALCulate:PERRor:MARKer<x>:PEAK:HIGHer`

Related Commands CALCulate:PERRor:MARKer<x>:PEAK:LOWer

Arguments None

Examples `CALCULATE:PERROR:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

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

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

Conditions Measurement views: Phase error versus Time

Group Calculate commands

Syntax `CALCulate:PERRor:MARKer<x>:PEAK:LEFT`

Related Commands CALCulate:PERRor:MARKer<x>:PEAK:RIGHt

Arguments None

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

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

Moves the selected marker to the next peak lower in amplitude on the Phase error versus Time trace.

Conditions Measurement views: Phase error versus Time

Group Calculate commands

Syntax `CALCulate:PERRor:MARKer<x>:PEAK:LOWer`

Related Commands CALCulate:PERRor:MARKer<x>:PEAK:HIGHer

Arguments None

Examples `CALCULATE:PERROR:MARKER1:PEAK:LOWER` moves Marker 1 (M1) lower in amplitude to the next peak on the trace.

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

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

Conditions Measurement views: Phase error versus Time

Group Calculate commands

Syntax `CALCulate:PERRor:MARKer<x>:PEAK:RIGHt`

Related Commands CALCulate:PERRor:MARKer<x>:PEAK:LEFT

Arguments None

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

CALCulate:PERRor:MARKer<x>:X

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

Conditions Measurement views: Phase error versus Time

Group Calculate commands

Syntax
```
CALCulate:PERRor:MARKer<x>:X <value>
CALCulate:PERRor:MARKer<x>:X?
```

Arguments <value>::=<NRf> specifies the horizontal position of the marker.
Range: (analysis offset) to [(analysis offset) + (analysis length)].
The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples CALCULATE:PERROR:MARKER1:X 1.5u places Marker 1 (M1) at 1.5 µs on the trace.

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

Queries the marker amplitude of the selected marker in the Phase error versus Time measurement.

Conditions Measurement views: Phase error versus Time

Group Calculate commands

Syntax
```
CALCulate:PERRor:MARKer<x>:Y?
```

Related Commands CALCulate:PERRor:MARKer<x>:X

Arguments None

Returns <NRf> Marker amplitude of the selected marker in degrees.

Examples CALCULATE:PERROR:MARKER1:Y? might return 21.04, indicating Marker 1 (M1) is at 21.04 °.

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Phase versus Time

Group Calculate commands

Syntax `CALCulate:PHVTime:MARKer<x>:DELTa:X?`

Related Commands CALCulate:PHVTime:MARKer<x>:DELTa:Y?

Arguments None

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

Examples `CALCULATE:PHVTIME:MARKER1:DELTA:X?` might return `38.0E-9`, indicating that the delta marker time is 38.0 ns.

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

Returns the delta marker phase for the selected marker in the Phase versus Time measurement.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Phase versus Time

Group Calculate commands

Syntax `CALCulate:PHVTime:MARKer<x>:DELTa:Y?`

Related Commands CALCulate:PHVTime:MARKer<x>:DELTa:X?

Arguments None

Returns <NRf> Delta marker phase for the selected marker in degrees.

Examples CALCULATE:PHVTIME:MARKER1:DELTA:Y? might return 162.38, indicating that the delta marker phase is 162.38 °.

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

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

Conditions Measurement views: Phase versus Time

Group Calculate commands

Syntax CALCulate:PHVTime:MARKer<x>:MAXimum

Arguments None

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

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

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

Conditions Measurement views: Phase versus Time

Group Calculate commands

Syntax CALCulate:PHVTime:MARKer<x>:PEAK:HIGHer

Related Commands CALCulate:PHVTime:MARKer<x>:PEAK:LOWer

Arguments None

Examples `CALCULATE:PHVTIME:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

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

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

Conditions Measurement views: Phase versus Time

Group Calculate commands

Syntax `CALCulate:PHVTime:MARKer<x>:PEAK:LEFT`

Related Commands CALCulate:PHVTime:MARKer<x>:PEAK:RIGHt

Arguments None

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

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

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

Conditions Measurement views: Phase versus Time

Group Calculate commands

Syntax `CALCulate:PHVTime:MARKer<x>:PEAK:LOWer`

Related Commands CALCulate:PHVTime:MARKer<x>:PEAK:HIGHer

Arguments None

Examples `CALCULATE:PHVTIME:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

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

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

Conditions Measurement views: Phase versus Time

Group Calculate commands

Syntax `CALCulate:PHVTime:MARKer<x>:PEAK:RIGHt`

Related Commands CALCulate:PHVTime:MARKer<x>:PEAK:LEFT

Arguments None

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

CALCulate:PHVTime:MARKer<x>:X

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

Conditions Measurement views: Phase versus Time

Group Calculate commands

Syntax
```
CALCulate:PHVTime:MARKer<x>:X <value>
CALCulate:PHVTime:MARKer<x>:X?
```

Related Commands CALCulate:PHVTime:MARKer<x>:Y?

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

Examples `CALCULATE:PHVTIME:MARKER1:X 1.5US` places Marker 1 (M1) at 1.5 µs on the trace.

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

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

Conditions Measurement views: Phase versus Time

Group Calculate commands

Syntax `CALCulate:PHVTime:MARKer<x>:Y?`

Related Commands CALCulate:PHVTime:MARKer<x>:X

Arguments None

Returns `<NRf>` Marker amplitude of the selected marker in degrees.

Examples `CALCULATE:PHVTIME:MARKER1:Y?` might return `-18.435`, indicating Marker 1 (M1) is at -18.435 °.

CALCulate:PULSe:STATistics:FFT:INDicator:X

Sets or queries the frequency in Hz for the X indicator. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to FFT.

Conditions Measurement views: Pulse statistics

Group Calculate commands

Syntax
```
CALCulate:PULSe:STATistics:FFT:INDicator:X <value>
CALCulate:PULSe:STATistics:FFT:INDicator:X?
```

Related Commands CALCulate:PULSe:STATistics:FFT:INDicator:Y?

Arguments <value>::=<NRf> specifies the indicator X value in Hz.

Examples CALCULATE:PULSE:STATISTICS:FFT:INDICATOR:X 1000 sets the FFT indicator X value to 1 kHz.

CALCulate:PULSe:STATistics:FFT:INDicator:Y? (Query Only)

Returns the frequency level in dB for the Y indicator. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to FFT.

Conditions Measurement views: Pulse statistics

Group Calculate commands

Syntax CALCulate:PULSe:STATistics:FFT:INDicator:Y?

Related Commands CALCulate:PULSe:STATistics:FFT:INDicator:X

Returns <NRf> the indicator Y value in dB.

Examples CALCULATE:PULSE:STATISTICS:FFT:INDICATOR:Y? might return -34.5 indicating that the FFT indicator is at -34.5 dB.

CALCulate:PULSe:STATistics:HISTogram:INDicator:X

Sets or queries the histogram indicator X value. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to histogram.

Conditions Measurement views: Pulse statistics

Group Calculate commands

Syntax
```
CALCulate:PULSe:STATistics:HISTogram:INDicator:X <value>
CALCulate:PULSe:STATistics:HISTogram:INDicator:X?
```

Related Commands CALCulate:PULSe:STATistics:HISTogram:INDicator:Y?

Arguments <value>::=<NRf> specifies the indicator X value.

Examples CALCULATE:PULSE:STATISTICS:HISTOGRAM:INDICATOR:X 0.3 sets the histogram indicator X value to 0.3.

CALCulate:PULSe:STATistics:HISTogram:INDicator:Y? (Query Only)

Queries the Histogram indicator Y value. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to Histogram.

Conditions Measurement views: Pulse statistics

Group Calculate commands

Syntax CALCulate:PULSe:STATistics:HISTogram:INDicator:Y?

Related Commands CALCulate:PULSe:STATistics:HISTogram:INDicator:X

Returns <NRf> indicates the indicator Y value.

Examples CALCULATE:PULSE:STATISTICS:HISTOGRAM:INDICATOR:Y? might return 20 to indicate 20 counts in the histogram bin.

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

Returns the delta marker time for the selected marker on the pulse trace. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to Time Trend.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Pulse statistics

Group Calculate commands

Syntax CALCulate:PULSe:STATistics:MARKer<x>:DELTa:X?

Related Commands DISPlay:PULSe:STATistics:PLOT

CALCulate:PULSe:STATistics:MARKer<x>:DELTa:Y?

Returns <NRf> Delta marker time in seconds for the selected marker.

Examples CALCULATE:PULSE:STATISTICS:MARKER1:DELTA:X? might return 3.01e-3, indicating that the delta marker time is 3.01 ms.

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

Returns the delta marker amplitude for the selected marker on the pulse trace. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to Time Trend.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Pulse statistics

Group Calculate commands

Syntax CALCulate:PULSe:STATistics:MARKer<x>:DELTa:Y?

Related Commands DISPlay:PULSe:STATistics:PLOT

CALCulate:PULSe:STATistics:MARKer<x>:DELTa:X?

Returns <NRf> Delta marker amplitude for the selected marker, in the unit of the current measurement.

Examples CALCULATE:PULSE:STATISTICS:MARKER1:DELTA:Y? might return 2.345, indicating that the delta marker amplitude is 2.345 dB.

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

Moves the selected marker to the highest peak on the pulse statistics trace. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to Time Trend.

Conditions Measurement views: Pulse statistics

Group Calculate commands

Syntax `CALCulate:PULSe:STATistics:MARKer<x>:MAXimum`

Arguments None

Examples `CALCULATE:PULSE:STATISTICS:MARKER1:MAXIMUM` moves Marker 1 (M1) to the highest peak on the trace.

CALCulate:PULSe:STATistics:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on the pulse statistics trace. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to Time Trend.

Conditions Measurement views: Pulse statistics

Group Calculate commands

Syntax `CALCulate:PULSe:STATistics:MARKer<x>:PEAK:HIGHer`

Related Commands CALCulate:PULSe:STATistics:MARKer<x>:PEAK:LOWer

Arguments None

Examples `CALCULATE:PULSE:STATISTICS:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

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

Moves the selected marker to the next peak to the left on the pulse statistics trace. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to Time Trend.

Conditions Measurement views: Pulse statistics

Group Calculate commands

Syntax CALCulate:PULSe:STATistics:MARKer<x>:PEAK:LEFT

Related Commands CALCulate:PULSe:STATistics:MARKer<x>:PEAK:RIGHt

Arguments None

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

CALCulate:PULSe:STATistics:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the pulse statistics trace. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to Time Trend.

Conditions Measurement views: Pulse statistics

Group Calculate commands

Syntax CALCulate:PULSe:STATistics:MARKer<x>:PEAK:LOWer

Related Commands CALCulate:PULSe:STATistics:MARKer<x>:PEAK:HIGHer

Arguments None

Examples CALCULATE:PULSE:STATISTICS:MARKER1:PEAK:LOWER moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

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

Moves the selected marker to the next peak to the right on the pulse statistics trace. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to Time Trend.

Conditions Measurement views: Pulse statistics

Group Calculate commands

Syntax `CALCulate:PULSe:STATistics:MARKer<x>:PEAK:RIGHt`

Related Commands CALCulate:PULSe:STATistics:MARKer<x>:PEAK:LEFT

Arguments None

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

CALCulate:PULSe:STATistics:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the pulse statistics view. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to Time Trend.

Conditions Measurement views: Pulse statistics

Group Calculate commands

Syntax
```
CALCulate:PULSe:STATistics:MARKer<x>:X <value>
CALCulate:PULSe:STATistics:MARKer<x>:X?
```

Related Commands CALCulate:PULSe:STATistics:MARKer<x>:Y?

Arguments `<value>::=<NRf>` specifies the horizontal position of the marker.

Examples `CALCULATE:PULSE:STATISTICS:MARKER1:X 3.01e-3` places Marker 1 (M1) at 3.01 ms on the trace.

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

Queries the marker amplitude of the selected marker in the pulse statistics view. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to Time Trend.

Conditions Measurement views: Pulse statistics

Group Calculate commands

Syntax `CALCulate:PULSe:STATistics:MARKer<x>:Y?`

Related Commands CALCulate:PULSe:STATistics:MARKer<x>:X

Arguments None

Returns `<NRf>` Amplitude of the specified marker in the unit of the current measurement.

Examples `CALCULATE:PULSE:STATISTICS:MARKER1:Y?` might return `-28.173`, indicating Marker 1 (M1) is at -28.173 dB.

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

Returns the delta marker time for the selected marker on the pulse trace.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Pulse trace

Group Calculate commands

Syntax `CALCulate:PULSe:TRACe:MARKer<x>:DELTa:X?`

Related Commands CALCulate:PULSe:TRACe:MARKer<x>:DELTa:Y?

Arguments None

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

Examples `CALCULATE:PULSE:TRACE:MARKER1:DELTA:X?` might return `38.0E-9`, indicating that the delta marker time is 38.0 ns.

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Pulse trace

Group Calculate commands

Syntax `CALCulate:PULSe:TRACe:MARKer<x>:DELTa:Y?`

Related Commands CALCulate:PULSe:TRACe:MARKer<x>:DELTa:X?

Arguments None

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

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

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

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

Conditions Measurement views: Pulse trace

Group Calculate commands

Syntax CALCulate:PULSe:TRACe:MARKer<x>:MAXimum

Arguments None

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

CALCulate:PULSe:TRACe:MARKer<x>:PEAK:HIGHer (No Query Form)

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

Conditions Measurement views: Pulse trace

Group Calculate commands

Syntax CALCulate:PULSe:TRACe:MARKer<x>:PEAK:HIGHer

Related Commands CALCulate:PULSe:TRACe:MARKer<x>:PEAK:LOWer

Arguments None

Examples `CALCULATE:PULSE:TRACE:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

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

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

Conditions Measurement views: Pulse trace

Group Calculate commands

Syntax `CALCulate:PULSe:TRACe:MARKer<x>:PEAK:LEFT`

Related Commands CALCulate:PULSe:TRACe:MARKer<x>:PEAK:RIGHt

Arguments None

Examples `CALCULATE:PULSE:TRACE:MARKER1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:PULSe:TRACe:MARKer<x>:PEAK:LOWer (No Query Form)

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

Conditions Measurement views: Pulse trace

Group Calculate commands

Syntax `CALCulate:PULSe:TRACe:MARKer<x>:PEAK:LOWer`

Related Commands CALCulate:PULSe:TRACe:MARKer<x>:PEAK:HIGHer

Arguments None

Examples `CALCULATE:PULSE:TRACE:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

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

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

Conditions Measurement views: Pulse trace

Group Calculate commands

Syntax `CALCulate:PULSe:TRACe:MARKer<x>:PEAK:RIGHt`

Related Commands CALCulate:PULSe:TRACe:MARKer<x>:PEAK:LEFT

Arguments None

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

CALCulate:PULSe:TRACe:MARKer<x>:X

Sets or queries the horizontal position of the selected marker in the pulse trace view.

Conditions Measurement views: Pulse trace

Group Calculate commands

Syntax
```
CALCulate:PULSe:TRACe:MARKer<x>:X <value>
CALCulate:PULSe:TRACe:MARKer<x>:X?
```

Related Commands CALCulate:PULSe:TRACe:MARKer<x>:Y?

Arguments `<value>::=<NRf>` specifies the horizontal position of the marker.

NOTE. *To set the horizontal position, the marker must be on the trace of the pulse selected using the DISPlay:PULSe:SELect:NUMBer command. You cannot put the marker out of the horizontal range of the pulse.*

Examples `CALCULATE:PULSE:TRACE:MARKER1:X 1.5us` places Marker 1 (M1) at 1.5 µs on the trace.

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

Queries the marker amplitude of the selected marker in the pulse trace view.

Conditions Measurement views: Pulse trace

Group Calculate commands

Syntax `CALCulate:PULSe:TRACe:MARKer<x>:Y?`

Related Commands CALCulate:PULSe:TRACe:MARKer<x>:X

Arguments None

Returns `<NRf>` Marker amplitude of the selected marker.

Examples `CALCULATE:PULSE:TRACE:MARKER1:Y?` might return `-28.86`, indicating Marker 1 (M1) is at -28.86 dBm.

CALCulate:SEARch:LIMit:FAIL? (Query Only)

Queries whether the waveform cuts across the limit in the search operation.

Conditions Measurement views: All

Group Calculate commands

Syntax `CALCulate:SEARch:LIMit:FAIL?`

Arguments None

Returns `{ 0 | 1 }`

0 represents Pass, indicating that the waveform does not cut across the limit.

1 represents Fail, indicating that the waveform cuts across the limit.

Examples CALCULATE:SEARCH:LIMIT:FAIL? might return 1, indicating that the waveform cuts across the limit (Fail).

CALCulate:SEARch:LIMit:MATCh:BEEP[:STATe]

Determines whether or not to beep on match during run or replay in the search operation.

Conditions Measurement views: All

Group Calculate commands

Syntax
```
CALCulate:SEARch:LIMit:MATCh:BEEP[:STATe] { OFF | ON | 0 | 1 }
CALCulate:SEARch:LIMit:MATCh:BEEP[:STATe]?
```

Arguments OFF or 0 disables to beep on match.

ON or 1 enables to beep on match.

Examples CALCULATE:SEARCH:LIMIT:MATCH:BEEP:STATE ON enables to beep on match.

CALCulate:SEARch:LIMit:MATCh:SACQuire[:STATe]

Determines whether or not to stop acquiring data on match during run or replay in the search operation.

Conditions Measurement views: All

Group Calculate commands

Syntax
```
CALCulate:SEARch:LIMit:MATCh:SACQuire[:STATe] { OFF | ON | 0 | 1 }
CALCulate:SEARch:LIMit:MATCh:SACQuire[:STATe]?
```

Arguments `OFF` or `0` disables to stop acquiring data on match.

`ON` or `1` enables to stop acquiring data on match.

Examples `CALCULATE:SEARCH:LIMIT:MATCH:SACQUIRE:STATE ON` enables to stop acquiring data on match.

CALCulate:SEARch:LIMit:MATCh:SDATa[:STATe]

Determines whether or not to save automatically (AutoSave) acquisition data on match during run in the search operation.

Conditions Measurement views: All

Group Calculate commands

Syntax
```
CALCulate:SEARch:LIMit:MATCh:SDATa[:STATe] { OFF | ON | 0 | 1 }
CALCulate:SEARch:LIMit:MATCh:SDATa[:STATe]?
```

Arguments `OFF` or `0` disables the AutoSave.

`ON` or `1` enables to save acquisition data automatically on match.
The data is saved to a file with the name:

<name>-yyyy.mm.dd.hh.mm.ss.sss.tiq

Where
<name> is the file name that was last specified.
yyyy.mm.dd and hh.mm.ss.sss represent date and 24 hour time.
The file extension is .tiq.
Example: SAVED-2007.03.20.12.34.567.tiq

For the directory of file, refer to *Specifying the File* (See page 2-60.) in the MMEMory command section.

Examples `CALCULATE:SEARCH:LIMIT:MATCH:SDATA:STATE ON` enables to save acquisition data automatically on match.

CALCulate:SEARch:LIMit:MATCh:SPICture[:STATe]

Determines whether or not to save automatically (AutoSave) the whole screen on match during run in the search operation.

Conditions Measurement views: All

Group Calculate commands

Syntax
```
CALCulate:SEARch:LIMit:MATCh:SPICture[:STATe] { OFF | ON | 0 | 1 }
CALCulate:SEARch:LIMit:MATCh:SPICture[:STATe]?
```

Arguments OFF or 0 disables the AutoSave.

ON or 1 enables to save the whole screen automatically on match.
The picture is saved to a file with the name:

<name>-yyyy.mm.dd.hh.mm.ss.sss.png

Where
<name> is the file name that was last specified.
yyyy.mm.dd and hh.mm.ss.sss represent date and 24 hour time.
The file extension is .png.
Example: SAVED-2007.03.20.12.34.567.png

For the directory of file, refer to *Specifying the File* (See page 2-60.) in the MMEMory command section.

Examples CALCULATE:SEARCH:LIMIT:MATCH:SPICTURE:STATE ON enables to save the whole screen automatically on match.

CALCulate:SEARch:LIMit:MATCh:STRace[:STATe]

Determines whether or not to save automatically (AutoSave) the spectrum trace on match during run in the search operation.

Conditions Measurement views: All

Group Calculate commands

Syntax
```
CALCulate:SEARch:LIMit:MATCh:STRace[:STATe] { OFF | ON | 0 | 1 }
CALCulate:SEARch:LIMit:MATCh:STRace[:STATe]?
```

Arguments OFF or 0 disables the AutoSave.

ON or 1 enables to save the spectrum trace automatically on match.

The trace is saved to a file with the name:

<name>-yyyy.mm.dd.hh.mm.ss.sss.Specan

Where
<name> is the file name that was last specified.
yyyy.mm.dd and hh.mm.ss.sss represent date and 24 hour time.
The file extension is .Specan.
Example: SAVED-2007.03.20.12.34.567.Specan

For the directory of file, refer to *Specifying the File* (See page 2-60.) in the MMEMory command section.

Examples `CALCULATE:SEARCH:LIMIT:MATCH:STRACE:STATE ON` enables to save the spectrum trace automatically on match.

CALCulate:SEARch:LIMit:OPERation

Sets or queries the search condition.

Conditions Measurement views: All

Group Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:OPERation { LT | GT | IMASk | OMASk }
CALCulate:SEARch:LIMit:OPERation?
```

Related Commands CALCulate:SEARch:LIMit:OPERation:SLIMit

Arguments The following table lists the arguments.

Argument	Meaning
LT	The data is less than the limit value.
GT	The data is greater than the limit value.
IMASk	The data is inside the limit mask.
OMASk	The data is outside the limit mask.

You can select the data using the CALCulate:SEARch:LIMit:OPERation:FEED command.

You can set the limit value using the CALCulate:SEARch:LIMit:OPERation:SLIMit command.

You can store and load the limit mask using the CALCulate:SEARch:LIMit:OPERation:MASK:STORE and CALCulate:SEARch:LIMit:OPERation:MASK:LOAD commands.

Examples `CALCULATE:SEARCH:LIMIT:OPERATIONGT` selects "the data is greater than the limit value" for the search condition.

CALCulate:SEARch:LIMit:OPERation:FEED

Sets or queries the data flow to be fed in the search operation.

Conditions Measurement views: All

Group Calculate commands

Syntax

```
CALCulate:SEARch:LIMit:OPERation:FEED <view>,<trace>
CALCulate:SEARch:LIMit:OPERation:FEED?
```

Arguments `<view>::=<string>` and `<trace>::=<string>` are listed in the following table.

Source data of the search operation

<view>	<trace>	Meaning
"Spectrum"	"Trace 1" [1]	Trace 1 in the Spectrum view.
	"Trace 2" [1]	Trace 2 in the Spectrum view.
	"Trace 3" [1]	Trace 3 in the Spectrum view.
	"Math Trace"	Math trace in the Spectrum view.
	"Spectrogram Trace"	Spectrogram trace in the Spectrum view.

[1] There is a space character between Trace and the number.

Examples `CALCULATE:SEARCH:LIMIT:OPERATION:FEED"Spectrum","Trace 1"` selects the Trace 1 in the Spectrum measurement view for the search operation.

`CALCULATE:SEARCH:LIMIT:OPERATION:FEED?` might return `"Spectrum","Math Trace"`, indicating that the math trace is used as the source data in the search operation.

CALCulate:SEARch:LIMit:OPERation:MASK:LOAD (No Query Form)

Loads the specified limit mask file for the search operation.

Conditions Measurement views: All

Group Calculate commands

Syntax `CALCulate:SEARch:LIMit:OPERation:MASK:LOAD <file_name>`

Arguments `<file_name>::=<string>` specifies the file to load the limit mask from. The file extension is .lmt. You can omit the extension.

For the directory of file, refer to *Specifying the File* (See page 2-60.) in the MMEMory command section.

Examples `CALCULATE:SEARCH:LIMIT:OPERATION:MASK:LOAD"Limit1"` loads the limit mask from the *Limit1.lmt* file.

CALCulate:SEARch:LIMit:OPERation:MASK:STORE (No Query Form)

Stores the limit mask to a specified file in the search operation.

Conditions Measurement views: All

Group Calculate commands

Syntax `CALCulate:SEARch:LIMit:OPERation:MASK:STORE <file_name>`

Arguments `<file_name>::=<string>` specifies the file to store the limit mask to. The file extension is .lmt. You can omit the extension.

For the directory of file, refer to *Specifying the File* (See page 2-60.) in the MMEMory command section.

Examples `CALCULATE:SEARCH:LIMIT:OPERATION:MASK:STORE"Limit1"` stores the limit mask to the *Limit1.lmt* file.

CALCulate:SEARch:LIMit:OPERation:SLIMit

Sets or queries the limit value (in dB) in the search operation.

Conditions Measurement views: All

Group Calculate commands

Syntax
```
CALCulate:SEARch:LIMit:OPERation:SLIMit <value>
CALCulate:SEARch:LIMit:OPERation:SLIMit?
```

Related Commands CALCulate:SEARch:LIMit:OPERation

Arguments <value>::=<NRf> specifies the limit value in the search operation.
Range: -100 to +100 dBm.

Examples CALCULATE:SEARCH:LIMIT:OPERATION:SLIMIT-20 sets the limit value to -20 dBm.

CALCulate:SEARch:LIMit:REPort:DATA? (Query Only)

Returns the frequency range(s) that satisfy the search condition.

Conditions Measurement views: All

Group Calculate commands

Syntax
```
CALCulate:SEARch:LIMit:REPort:DATA?
```

Arguments None

Returns
```
<num_range>,<range(1)>,<range(2)>,...,<range(n)>
```

Where
<num_range>::=<NR1> is the number of ranges that satisfy the condition.
<range(n)>::="<lower_freq(n)>, <upper_freq(n)>" (string) represents the n^{th} frequency range that satisfy the search condition in ascending

order. <lower_freq(n)> and <upper_freq(n)> are the lower and upper frequencies of the range #n, respectively.

Examples CALCULATE:SEARCH:LIMIT:REPORT:DATA? might return 2,"1.4800E+9, 1.5001E+9","1.5002E+9, 1.5200E+9", indicating that the search condition is satisfied in these two ranges 1.48 to 1.5001 GHz and 1.5002 to 1.52 GHz.

CALCulate:SEARch:LIMit:REPort:POINts? (Query Only)

Returns the number of frequency range(s) that satisfy the search condition.

Conditions Measurement views: All

Group Calculate commands

Syntax CALCulate:SEARch:LIMit:REPort:POINts?

Arguments None

Returns <number>::=<NR1> represents the number of frequency range(s) that satisfy the search condition.

Examples CALCULATE:SEARCH:LIMIT:REPORT:POINTS? might return 5, indicating that five ranges satisfy the search condition.

CALCulate:SEARch:LIMit:STATe

Sets or queries the search function (enabled or disabled).

Conditions Measurement views: All

Group Calculate commands

Syntax
```
CALCulate:SEARch:LIMit:STATe { OFF | ON | 0 | 1 }
CALCulate:SEARch:LIMit:STATe?
```

Arguments `OFF` or `0` disables the search function.

`ON` or `1` enables the search function.

Examples `CALCULATE:SEARCH:LIMIT:STATEON` enables the search function.

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

Returns the delta marker frequency for the selected marker on the Spectral Emissions Mask trace.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Spectral Emissions Mask

Group Calculate commands

Syntax `CALCulate:SEM:MARKer<x>:DELTa:X?`

Arguments None

Returns `<NRf>` Delta marker frequency for the selected marker.

Examples `CALCulate:SEM:MARKer1:DELTa:X?` might return `1.28E+6`, indicating that the delta marker frequency is 1.28 MHz.

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

Returns the delta marker amplitude for the selected marker on the Spectral Emissions Mask trace.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Spectral Emissions Mask

Group Calculate commands

Syntax CALCulate:SEM:MARKer<x>:DELTa:Y?

Arguments None

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

Examples CALCulate:SEM:MARKer1:DELTa:Y? might return 23.45, indicating that the delta marker amplitude is 23.45 dB.

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

Moves the selected marker to the highest peak on the Spectral Emissions Mask trace.

Conditions Measurement views: Spectral Emissions Mask

Group Calculate commands

Syntax CALCulate:SEM:MARKer<x>:MAXimum

Arguments None

Examples CALCulate:SEM:MARKer1:MAXimum moves Marker 1 (M1) to the highest peak on the trace.

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

Moves the selected marker to the next peak higher in amplitude on the Spectral Emissions Mask trace.

Conditions Measurement views: Spectral Emissions Mask

Group Calculate commands

Syntax CALCulate:SEM:MARKer<x>:PEAK:HIGHer

Arguments None

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

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

Moves the selected marker to the next peak to the left on the Spectral Emissions Mask trace.

Conditions Measurement views: Spectral Emissions Mask

Group Calculate commands

Syntax `CALCulate:SEM:MARKer<x>:PEAK:LEFT`

Arguments None

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

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

Moves the selected marker to the next peak lower in amplitude on the Spectral Emissions Mask trace.

Conditions Measurement views: Spectral Emissions Mask

Group Calculate commands

Syntax `CALCulate:SEM:MARKer<x>:PEAK:LOWer`

Arguments None

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

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

Moves the selected marker to the next peak to the right on the Spectral Emissions Mask trace.

Conditions Measurement views: Spectral Emissions Mask

Group Calculate commands

Syntax `CALCulate:SEM:MARKer<x>:PEAK:RIGHt`

Arguments None

Examples CALCulate:SEM:MARKer1:PEAK:RIGHt moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:SEM:MARKer<x>[:SET]:CENTer (No Query Form)

Moves the specified marker to the center frequency.

Conditions Measurement views: Spectral Emissions Mask

Group Calculate commands

Syntax `CALCulate:SEM:MARKer<x>[:SET]:CENTer`

Arguments None

Examples `CALCulate:SEM:MARKer1[:SET]:CENTer` moves Marker 1 (M1) to the center frequency.

CALCulate:SEM:MARKer<x>:X

Sets or queries the horizontal position of the selected marker on the Spectral Emissions Mask trace.

Conditions Measurement views: Spectral Emissions Mask

Group Calculate commands

Syntax
```
CALCulate:SEM:MARKer<x>:X <value>
CALCulate:SEM:MARKer<x>:X?
```

Arguments <value>::=<NRf> specifies the horizontal position of the marker.

Examples CALCULATE:SEM:MARKER1:X 800MHz places Marker 1 (M1) at 800 MHz on the spectrum trace.

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

Queries the marker amplitude of the selected marker on the Spectral Emissions Mask trace.

Conditions Measurement views: Spectral Emissions Mask

Group Calculate commands

Syntax
```
CALCulate:SEM:MARKer<x>:Y?
```

Arguments None

Returns <NRf> Marker amplitude of the selected marker.

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

CALCulate:SGRam:MARKer<x>:DELTa:X:FREQuency?

Returns the delta marker frequency for the selected marker in the spectrogram.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Spectrogram

Group Calculate commands

Syntax `CALCulate:SGRam:MARKer<x>:DELTa:X:FREQuency?`

Related Commands CALCulate:SGRam:MARKer<x>:DELTa:Y?

Arguments None

Returns `<NRf>` Delta marker frequency for the selected marker.

Examples `CALCULATE:SGRAM:MARKER1:DELTA:X:FREQUENCY?` might return `5.95E+6`, indicating that the delta marker frequency is 5.95 MHz.

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

Returns the delta marker time for the selected marker in the spectrogram.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Spectrogram

Group Calculate commands

Syntax `CALCulate:SGRam:MARKer<x>:DELTa:X[:TIME]?`

Related Commands CALCulate:SGRam:MARKer<x>:DELTa:Y?

Arguments None

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

Examples CALCULATE:SGRAM:MARKER1:DELTA:X:TIME? might return -1.84E-3, indicating that the delta marker time is -1.84 ms.

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

Returns the delta marker amplitude for the selected marker in the spectrogram.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Spectrogram

Group Calculate commands

Syntax CALCulate:SGRam:MARKer<x>:DELTa:Y?

Related Commands CALCulate:SGRam:MARKer<x>:DELTa:X[:TIME]?

Arguments None

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

Examples CALCULATE:SGRAM:MARKER1:DELTA:Y? might return -8.45, indicating that the delta marker amplitude is -8.45 dB.

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

Moves the selected marker to the highest peak on a line in the spectrogram. The line is selected using the TRACe:SGRam:SELect:LINE command.

Conditions Measurement views: Spectrogram

Group Calculate commands

Syntax `CALCulate:SGRam:MARKer<x>:MAXimum`

Arguments None

Examples `CALCULATE:SGRAM:MARKER1:MAXIMUM` moves Marker 1 (M1) to the highest peak on the line.

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

Moves the selected marker to the next peak higher in amplitude on a line in the spectrogram. The line is selected using the TRACe:SGRam:SELect:LINE command.

Conditions Measurement views: Spectrogram

Group Calculate commands

Syntax `CALCulate:SGRam:MARKer<x>:PEAK:HIGHer`

Related Commands CALCulate:SGRam:MARKer<x>:PEAK:LOWer

Arguments None

Examples `CALCULATE:SGRAM:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the line.

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

Moves the selected marker to the next peak to the left on a line in the spectrogram. The line is selected using the TRACe:SGRam:SELect:LINE command.

Conditions Measurement views: Spectrogram

Group Calculate commands

Syntax `CALCulate:SGRam:MARKer<x>:PEAK:LEFT`

Related Commands CALCulate:SGRam:MARKer<x>:PEAK:RIGHt

Arguments None

Examples `CALCULATE:SGRAM:MARKER1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the line.

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

Moves the selected marker to the next peak lower in amplitude on a line in the spectrogram. The line is selected using the TRACe:SGRam:SELect:LINE command.

Conditions Measurement views: Spectrogram

Group Calculate commands

Syntax `CALCulate:SGRam:MARKer<x>:PEAK:LOWer`

Related Commands CALCulate:SGRam:MARKer<x>:PEAK:HIGHer

Arguments None

Examples `CALCULATE:SGRAM:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the line.

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

Moves the selected marker to the next peak to the right on a line in the spectrogram. The line is selected using the TRACe:SGRam:SELect:LINE command.

Conditions Measurement views: Spectrogram

Group Calculate commands

Syntax `CALCulate:SGRam:MARKer<x>:PEAK:RIGHt`

Related Commands CALCulate:SGRam:MARKer<x>:PEAK:LEFT

Arguments None

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

CALCulate:SGRam:MARKer<x>[:SET]:CENTer (No Query Form)

Sets the center frequency to the marker frequency in the spectrogram.

Conditions Measurement views: Spectrogram

Group Calculate commands

Syntax `CALCulate:SGRam:MARKer<x>[:SET]:CENTer`

Arguments None

Examples `CALCULATE:SGRAM:MARKER1:SET:CENTER` sets the center frequency to the Marker 1 frequency in the spectrogram.

CALCulate:SGRam:MARKer<x>:X:FREQuency

Sets or queries the marker frequency in the spectrogram.

Conditions Measurement views: Spectrogram

Group Calculate commands

Syntax
```
CALCulate:SGRam:MARKer<x>:X:FREQuency <value>
CALCulate:SGRam:MARKer<x>:X:FREQuency?
```

Related Commands CALCulate:SGRam:MARKer<x>:Y?

Arguments <value>::=<NRf> specifies the marker frequency.
Range: (center frequency) ± (span)/2.

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

CALCulate:SGRam:MARKer<x>:X[:TIME]

Sets or queries the marker time in the spectrogram.

Conditions Measurement views: Spectrogram

Group Calculate commands

Syntax
```
CALCulate:SGRam:MARKer<x>:X[:TIME] <value>
CALCulate:SGRam:MARKer<x>:X[:TIME]?
```

Related Commands CALCulate:SGRam:MARKer<x>:Y?

Arguments <value>::=<NRf> specifies the marker time.

Examples CALCULATE:SGRAM:MARKER1:X:TIME -234.5us places Marker 1 (M1) at -234.5 µs on the trace.

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

Queries the marker amplitude in the spectrogram.

Conditions Measurement views: Spectrogram

Group Calculate commands

Syntax
```
CALCulate:SGRam:MARKer<x>:Y?
```

Related Commands CALCulate:SGRam:MARKer<x>:X[:TIME]

Arguments None

Returns <NRf> Marker amplitude of the selected marker.

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

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Spectrum

Group Calculate commands

Syntax CALCulate:SPECtrum:MARKer<x>:DELTa:X?

Related Commands CALCulate:SPECtrum:MARKer<x>:DELTa:Y?

Arguments None

Returns <NRf> Delta marker frequency for the selected marker.

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

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Spectrum

Group Calculate commands

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

Related Commands CALCulate:SPECtrum:MARKer<x>:DELTa:X?

Arguments None

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

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

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

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

Conditions Measurement views: Spectrum

Group Calculate commands

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

Arguments None

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

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

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

Conditions Measurement views: Spectrum

Group Calculate commands

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

Related Commands CALCulate:SPECtrum:MARKer<x>:PEAK:LOWer

Arguments None

Examples `CALCULATE:SPECTRUM:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

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

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

Conditions Measurement views: Spectrum

Group Calculate commands

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

Related Commands CALCulate:SPECtrum:MARKer<x>:PEAK:RIGHt

Arguments None

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

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

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

Conditions Measurement views: Spectrum

Group Calculate commands

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

Related Commands CALCulate:SPECtrum:MARKer<x>:PEAK:HIGHer

Arguments None

Examples `CALCULATE:SPECTRUM:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

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

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

Conditions Measurement views: Spectrum

Group Calculate commands

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

Related Commands CALCulate:SPECtrum:MARKer<x>:PEAK:LEFT

Arguments None

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

CALCulate:SPECtrum:MARKer<x>:POWer:DENSity? (Query Only)

Returns the power density of the specified marker in the Spectrum measurement.

Conditions Measurement view: Spectrum

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid.

Group Calculate commands

Syntax `CALCulate:SPECtrum:MARKer<x>:POWer:DENSity?`

Related Commands CALCulate:SPECtrum:MARKer<x>:POWer:IDENSity

CALCulate:SPECtrum:MARKer<x>:POWer:INTEgrated?

Returns <NR2> = a floating point value without exponents. It is the power density of the selected marker in dBm/Hz.

Examples `CALCULATE:SPECTRUM:MARKER1:POWER:DENSITY?` might return -80.2412563552, indicating that the power density is -80.24 dBm/Hz.

CALCulate:SPECtrum:MARKer<x>:POWer:IDENSity

Returns the integrated power density reading between the MR (reference marker) and the selected marker in the Spectrum measurement.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid.

Conditions Measurement view: Spectrum

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid.

Group Calculate commands

Syntax `CALCulate:SPECtrum:MARKer<x>:POWer:IDENSity`

Related Commands CALCulate:SPECtrum:MARKer<x>:POWer:INTEgrated?

CALCulate:SPECtrum:MARKer<x>:POWer:DENSity?

Returns <NR2> = a floating point value without exponents. It is the integrated power density reading between MR (reference marker) and the selected marker in dBm/Hz.

Examples `CALCULATE:SPECTRUM:MARKER1:POWER:IDENSITY?` might return -78.5625446112, indicating that the integrated power density is -78.56 dBm/Hz.

CALCulate:SPECtrum:MARKer<x>:POWer:INTEgrated? (Query Only)

Returns the integrated power reading between the MR (reference marker) and the selected marker in the Spectrum measurement.

Conditions Measurement view: Spectrum

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid.

Group Calculate commands

Syntax `CALCulate:SPECtrum:MARKer<x>:POWer:INTEgrated?`

Related Commands CALCulate:SPECtrum:MARKer<x>:POWer:IDENSity

CALCulate:SPECtrum:MARKer<x>:POWer:DENSity?

Returns <NR2> = a floating point value without exponents. It is the integrated power reading between MR (reference marker) and the selected marker in dBm.

Examples `CALC:SPEC:MARK1:POW:INTE?` might return -12.4462022781, indicating the integrated power is -12.45 dBm.

CALCulate:SPECtrum:MARKer<x>[:SET]:CENTer (No Query Form)

Sets the center frequency to the marker frequency in the spectrum measurement.

Conditions Measurement views: Spectrum

Group Calculate commands

Syntax `CALCulate:SPECtrum:MARKer<x>[:SET]:CENTer`

Arguments None

Examples `CALCULATE:SPECTRUM:MARKER1:SET:CENTER` sets the center frequency to the marker frequency in the spectrum measurement.

CALCulate:SPECtrum:MARKer<x>:TRACe

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

Conditions Measurement views: Spectrum

Group Calculate commands

Syntax
```
CALCulate:SPECtrum:MARKer<x>:TRACe { TRACE1 | TRACE2 |
TRACE3 | TRACE4 | TRACE5 }
CALCulate:SPECtrum:MARKer<x>:TRACe?
```

Arguments `TRACE1` places the specified marker on Trace 1.

`TRACE2` places the specified marker on Trace 2.

`TRACE3` places the specified marker on n Trace 3.

`TRACE4` places the specified marker on Trace 4 (math trace).

`TRACE5` places the specified marker on Trace 5 (spectrogram).

Trace 1 to 3 can be defined as Normal, Average, Max Hold or Min Hold using the command.

Examples `CALCULATE:SPECTRUM:MARKER1:TRACE TRACE1` places Marker 1 (M1) on Trace 1.

CALCulate:SPECtrum:MARKer<x>:X

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

Conditions Measurement views: Spectrum

Group Calculate commands

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

Related Commands CALCulate:SPECtrum:MARKer<x>:Y?

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

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

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

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

Conditions Measurement views: Spectrum

Group Calculate commands

Syntax `CALCulate:SPECtrum:MARKer<x>:Y?`

Related Commands CALCulate:SPECtrum:MARKer<x>:X

Arguments None

Returns `<NRf>` Marker amplitude of the selected marker.

Examples `CALCULATE:SPECTRUM:MARKER1:Y?` might return `-34.28`, indicating Marker 1 (M1) is at -34.28 dBm.

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Spurious

Group Calculate commands

Syntax `CALCulate:SPURious:MARKer<x>:DELTa:X?`

Arguments None

Returns <NRf> Delta marker frequency for the selected marker.

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

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

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

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Spurious

Group Calculate commands

Syntax CALCulate:SPURious:MARKer<x>:DELTa:Y?

Arguments None

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

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

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

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

Conditions Measurement views: Spurious

Group Calculate commands

Syntax CALCulate:SPURious:MARKer<x>:MAXimum

Arguments None

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

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

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

Conditions Measurement views: Spurious

Group Calculate commands

Syntax `CALCulate:SPURious:MARKer<x>:PEAK:HIGHer`

Arguments None

Examples `CALCULATE:SPURIOUS:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

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

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

Conditions Measurement views: Spurious

Group Calculate commands

Syntax `CALCulate:SPURious:MARKer<x>:PEAK:LEFT`

Arguments None

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

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

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

Conditions Measurement views: Spurious

Group Calculate commands

Syntax `CALCulate:SPURious:MARKer<x>:PEAK:LOWer`

Arguments None

Examples `CALCULATE:SPURIOUS:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

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

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

Conditions Measurement views: Spurious

Group Calculate commands

Syntax `CALCulate:SPURious:MARKer<x>:PEAK:RIGHt`

Arguments None

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

CALCulate:SPURious:MARKer<x>[:SET]:CENTer (No Query Form)

Sets the center frequency to the marker frequency in the Spurious measurement.

Conditions Measurement views: Spurious

Group Calculate commands

Syntax `CALCulate:SPURious:MARKer<x>[:SET]:CENTer`

Arguments None

Examples `CALCULATE:SPURIOUS:MARKER1:SET:CENTER` sets the center frequency to the value at Marker 1.

CALCulate:SPURious:MARKer<x>:X

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

Conditions Measurement views: Spurious

Group Calculate commands

Syntax
```
CALCulate:SPURious:MARKer<x>:X <value>
CALCulate:SPURious:MARKer<x>:X?
```

Arguments `<value>::=<NRf>` specifies the horizontal position of the marker.
Range: Start to Stop frequency (left to right edge of the horizontal axis).

The start and stop frequencies are set using the DISPlay:SPURious:X[:SCALe]:STARt and DISPlay:SPURious:X[:SCALe]:STOP commands.

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

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

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

Conditions Measurement views: Spurious

Group Calculate commands

Syntax `CALCulate:SPURious:MARKer<x>:Y?`

Arguments None

Returns `<NRf>` Marker amplitude of the selected marker.

Examples `CALCULATE:SPURIOUS:MARKER1:Y?` might return `-34.28`, indicating Marker 1 (M1) is at -34.28 dBm.

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

Returns the delta marker time for the selected marker on the Trellis diagram trace.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Trellis diagram

Group Calculate commands

Syntax `CALCulate:TDIagram:MARKer<x>:DELTa:X[:TIME]?`

Related Commands CALCulate:TDIagram:MARKer<x>:DELTa:Y?

Arguments None

Returns `<NRf>` Delta marker time for the selected marker.
Use the [SENSe]:DDEMod:TIMe:UNITs command to select the time unit: symbols (default) or seconds.

Examples `CALCULATE:TDIAGRAM:MARKER1:DELTA:X:TIME?` might return `62.75`, indicating that the delta marker time is 62.75 symbols.

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

Returns the delta marker amplitude for the selected marker on the Trellis diagram trace.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Trellis diagram

Group Calculate commands

Syntax `CALCulate:TDIagram:MARKer<x>:DELTa:Y?`

Related Commands CALCulate:TDIagram:MARKer<x>:DELTa:X[:TIME]?

Arguments None

Returns `<NRf>` Delta marker amplitude for the selected marker in degrees.

Examples `CALCULATE:TDIAGRAM:MARKER1:DELTA:Y?` might return `-48.26`, indicating that the delta marker amplitude is -48.26°.

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

Moves the selected marker to the highest peak on the Trellis diagram trace.

Conditions Measurement views: Trellis diagram

Group Calculate commands

Syntax `CALCulate:TDIagram:MARKer<x>:MAXimum`

Arguments None

Examples `CALCULATE:TDIAGRAM:MARKER1:MAXIMUM?` moves Marker 1 (M1) to the highest peak on the trace.

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

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

Conditions Measurement views: Trellis diagram

Group Calculate commands

Syntax `CALCulate:TDIagram:MARKer<x>:PEAK:HIGHer`

Related Commands CALCulate:TDIagram:MARKer<x>:PEAK:LOWer

Arguments None

Examples `CALCULATE:TDIAGRAM:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

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

Moves the selected marker to the next peak to the left on the Trellis diagram trace.

Conditions Measurement views: Trellis diagram

Group Calculate commands

Syntax `CALCulate:TDIagram:MARKer<x>:PEAK:LEFT`

Related Commands CALCulate:TDIagram:MARKer<x>:PEAK:RIGHt

Arguments None

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

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

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

Conditions Measurement views: Trellis diagram

Group Calculate commands

Syntax `CALCulate:TDIagram:MARKer<x>:PEAK:LOWer`

Related Commands CALCulate:TDIagram:MARKer<x>:PEAK:HIGHer

Arguments None

Examples `CALCULATE:TDIAGRAM:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

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

Moves the selected marker to the next peak to the right on the Trellis diagram trace.

Conditions Measurement views: Trellis diagram

Group Calculate commands

Syntax `CALCulate:TDIagram:MARKer<x>:PEAK:RIGHt`

Related Commands CALCulate:TDIagram:MARKer<x>:PEAK:LEFT

Arguments None

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

CALCulate:TDIagram:MARKer<x>:X[:TIME]

Sets or queries the horizontal position (time) of the selected marker in the Trellis diagram measurement.

Conditions Measurement views: Trellis diagram

Group Calculate commands

Syntax
```
CALCulate:TDIagram:MARKer<x>:X[:TIME] <value>
CALCulate:TDIagram:MARKer<x>:X[:TIME]?
```

Related Commands CALCulate:TDIagram:MARKer<x>:Y?

Arguments `<value>::=<NRf>` specifies the horizontal position (time) of the marker. Use the [SENSe]:DDEMod:TIMe:UNITs command to select the time unit: symbols (default) or seconds.

Examples `CALCULATE:TDIAGRAM:MARKER1:X:TIME 38.5` places Marker 1 (M1) at 38.5 symbols on the trace.

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

Queries the marker amplitude of the selected marker in the Trellis diagram measurement.

Conditions Measurement views: Trellis diagram

Group Calculate commands

Syntax
```
CALCulate:TDIagram:MARKer<x>:Y?
```

Related Commands CALCulate:TDIagram:MARKer<x>:X[:TIME]

Arguments None

Returns `<NRf>` Marker amplitude of the selected marker in degrees.

Examples `CALCULATE:TDIAGRAM:MARKER1:Y?` might return `212.3`, indicating Marker 1 (M1) is at 212.3 °.

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

Returns the delta marker time for the selected marker on the time overview trace.

Conditions Measurement views: Time overview

Group Calculate commands

Syntax `CALCulate:TOVerview:MARKer<x>:DELTa:X?`

Related Commands CALCulate:TOVerview:MARKer<x>:DELTa:Y?

Arguments None

Returns `<NRf>` Delta marker time for the selected marker.

Examples `CALCULATE:TOVERVIEW:MARKER1:DELTA:X?` might return `38.0E-9`, indicating that the delta marker time is 38.0 ns.

CALCulate:TOVerview:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker amplitude for the selected marker on the time overview trace.

The parameter <x> = 1 to 4; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Time overview

Group Calculate commands

Syntax `CALCulate:TOVerview:MARKer<x>:DELTa:Y?`

Related Commands CALCulate:TOVerview:MARKer<x>:DELTa:X?

Arguments None

Returns <NRf> Delta marker amplitude for the selected marker.

Examples CALCULATE:TOVERVIEW:MARKER1:DELTA:Y? might return 23.45, indicating that the delta marker amplitude is 23.45 dB.

CALCulate:TOVerview:MARKer<x>:MAXimum (No Query Form)

Moves the selected marker to the highest peak on the time overview trace.

Conditions Measurement views: Time overview

Group Calculate commands

Syntax CALCulate:TOVerview:MARKer<x>:MAXimum

Arguments None

Examples CALCULATE:TOVERVIEW:MARKER1:MAXIMUM moves Marker 1 (M1) to the highest peak on the trace.

CALCulate:TOVerview:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the selected marker to the next peak higher in amplitude on the time overview trace.

Conditions Measurement views: Time overview

Group Calculate commands

Syntax CALCulate:TOVerview:MARKer<x>:PEAK:HIGHer

Related Commands CALCulate:TOVerview:MARKer<x>:PEAK:LOWer

Arguments None

Examples `CALCULATE:TOVERVIEW:MARKER1:PEAK:HIGHER` moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

CALCulate:TOVerview:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the selected marker to the next peak to the left on the time overview trace.

Conditions Measurement views: Time overview

Group Calculate commands

Syntax `CALCulate:TOVerview:MARKer<x>:PEAK:LEFT`

Related Commands CALCulate:TOVerview:MARKer<x>:PEAK:RIGHt

Arguments None

Examples `CALCULATE:TOVERVIEW:MARKER1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace.

CALCulate:TOVerview:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the selected marker to the next peak lower in amplitude on the time overview trace.

Conditions Measurement views: Time overview

Group Calculate commands

Syntax `CALCulate:TOVerview:MARKer<x>:PEAK:LOWer`

Related Commands CALCulate:TOVerview:MARKer<x>:PEAK:HIGHer

Arguments None

Examples `CALCULATE:TOVERVIEW:MARKER1:PEAK:LOWER` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALCulate:TOVerview:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the selected marker to the next peak to the right on the time overview trace.

Conditions Measurement views: Time overview

Group Calculate commands

Syntax `CALCulate:TOVerview:MARKer<x>:PEAK:RIGHt`

Related Commands CALCulate:TOVerview:MARKer<x>:PEAK:LEFT

Arguments None

Examples `CALCULATE:TOVERVIEW:MARKER1:PEAK:RIGHT` moves Marker 1 (M1) to the next peak to the right on the trace.

CALCulate:TOVerview:MARKer<x>:X

Sets or queries the horizontal position of the selected marker on the time overview trace.

Conditions Measurement views: Time overview

Group Calculate commands

Syntax
```
CALCulate:TOVerview:MARKer<x>:X <value>
CALCulate:TOVerview:MARKer<x>:X?
```

Related Commands CALCulate:TOVerview:MARKer<x>:Y?

Arguments `<value>::=<NRf>` specifies the horizontal position of the specified marker. Range: (analysis offset) to [(analysis offset) + (analysis length)].

Returns <NRf> is a floating point number that is the horizontal position of the specified marker.

Examples `CALCULATE:TOVERVIEW:MARKER1:X 1.5us` places Marker 1 (M1) at 1.5 µs on the trace.

CALCulate:TOVerview:MARKer<x>:Y? (Query Only)

Queries the marker amplitude of the selected marker in the time overview.

Conditions Measurement views: Time overview

Group Calculate commands

Syntax `CALCulate:TOVerview:MARKer<x>:Y?`

Related Commands CALCulate:TOVerview:MARKer<x>:X

Arguments None

Returns `<NRf>` Marker amplitude of the selected marker.

Examples `CALCULATE:TOVERVIEW:MARKER1:Y?` might return `-34.28`, indicating Marker 1 (M1) is at -34.28 dBm.

CALculate:TXGain:MARKer<x>:DELTa:X? (Query Only)

Returns the delta marker frequency for the specified marker on the trace.

Conditions Measurement view: Transmission Gain

The parameter <x> = 1 to 3; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group Calculate group

Syntax `CALculate:TXGain:MARKer<x>:DELTa:X?`

Returns <NRf> is the delta marker frequency for the specified marker.

Examples `CALculate:TXGain:MARKer1:DELTa:X?` might return 1.28E+6, indicating that the delta marker frequency is 1.28 MHz.

CALculate:TXGain:MARKer<x>:DELTa:Y? (Query Only)

Returns the delta marker amplitude for the specified marker on the trace.

Conditions Measurement view: Transmission Gain

The parameter <x> = 1 to 3; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group Calculate group

Syntax `CALculate:TXGain:MARKer<x>:DELTa:Y?`

Returns <NRf> is the delta marker amplitude for the specified marker.

Examples `CALculate:TXGain:MARKer1:DELTa:Y?` might return 23.45, indicating that the delta marker amplitude for Marker 1 (M1) is 23.45 dB.

CALculate:TXGain:MARKer<x>:MAXimum (No Query Form)

Moves the specified marker to the highest peak on the trace.

Conditions Measurement view: Transmission Gain

The parameter <x> = 1 to 3; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group Calculate group

Syntax `CALculate:TXGain:MARKer<x>:MAXimum`

Arguments None

Examples CALculate:TXGain:MARKer1:MAXimum moves Marker 1 (M1) to the highest peak on the trace.

CALculate:TXGain:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the specified marker to the next peak higher in amplitude on the trace.

Conditions Measurement view: Transmission Gain

The parameter <x> = 1 to 3; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group Calculate group

Syntax CALculate:TXGain:MARKer<x>:PEAK:HIGHer

Arguments None

Examples CALculate:TXGain:MARKer1:HIGHer moves Marker 1 (M1) to the next peak higher in amplitude on the trace.

CALculate:TXGain:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the specified marker to the next peak to the left on the trace.

Conditions Measurement view: Transmission Gain

The parameter <x> = 1 to 3; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group Calculate group

Syntax CALculate:TXGain:MARKer<x>:PEAK:LEFT

Arguments None

Examples `CALculate:TXGain:MARKer1:PEAK:LEFT` moves Marker 1 (M1) to the next peak to the left on the trace.

CALculate:TXGain:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the specified marker to the next peak lower in amplitude on the trace.

Conditions Measurement view: Transmission Gain

The parameter <x> = 1 to 3; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group Calculate group

Syntax `CALculate:TXGain:MARKer<x>:PEAK:LOWer`

Arguments None

Examples `CALculate:TXGain:MARKer1:PEAK:LOWer` moves Marker 1 (M1) to the next peak lower in amplitude on the trace.

CALculate:TXGain:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the specified marker to the next peak to the right on the trace.

Conditions Measurement view: Transmission Gain

The parameter <x> = 1 to 3; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group Calculate group

Syntax `CALculate:TXGain:MARKer<x>:PEAK:RIGHt`

Arguments None

Examples `CALculate:TXGain:MARKer1:PEAK:RIGHt` moves Marker 1 (M1) to the next peak to the right on the trace.

CALculate:TXGain:MARKer<x>:TRACe

Sets or queries the trace on which the specified marker is placed in the Transmission Gain display.

Conditions Measurement view: Transmission Gain

The parameter <x> = 1 to 3; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group Calculate group

Syntax

```
CALculate:TXGain:MARKer<x>:TRACe { TRACE1 | TRACE2 | TRACE3 }
CALculate:TXGain:MARKer<x>:TRACe?
```

Arguments TRACE 1 places the specified marker on Trace 1.

TRACE 2 places the specified marker on Trace 2.

TRACE 3 places the specified marker on Trace 3.

Returns See Arguments.

Examples `CALculate:TXGain:MARKer1:TRACe TRACE1` places Marker 1 (M1) on Trace 1.

CALculate:TXGain:MARKer<x>:X

Sets or queries the horizontal position for the specified marker on the trace.

Conditions Measurement view: Transmission Gain

The parameter <x> = 1 to 3; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group Calculate group

Syntax
```
CALculate:TXGain:MARKer<x>:X <value>
CALculate:TXGain:MARKer<x>:X ?
```

Arguments <value>::=<NRf> specifies the horizontal position of the specified marker.

Range: Start to Stop frequency (left to right edge of the horizontal axis). Using an out-of-range value causes an execution error (-222, "Data out of range").

Returns <NRf> is the horizontal position of the specified marker.

Examples `CALculate:TXGain:MARKer1:X 800E+6` places Marker 1 (M1) at 800 MHz on the trace.

CALculate:TXGain:MARKer<x>:Y? (Query Only)

Queries the marker amplitude of the specified marker on the trace.

Conditions Measurement view: Transmission Gain

The parameter <x> = 1 to 3; MARKer0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Group Calculate group

Syntax
```
CALculate:TXGain:MARKer<x>:Y?
```

Returns <NRf> is the amplitude at the specified marker position in dB.

Examples `CALculate:TXGain:MARKer1:Y` might return -23.45, indicating that the marker amplitude is -23.45 dB.

CALCulate:WLAN:CONSte:MARKer<x>:FREQuency

Sets or queries the frequency value of the marker on the WLAN Constellation display. The units are either Frequency in Hz or Subcarrier number. To set the units, use the command [SENSe]:WLAN:UNIT:FREQuency.

Note that FREQuency is not an actual measurement of frequency, but is the subcarrier offset within the analyzed packet burst.

<x>, which is optional, is the desired marker to use. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Commanding a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax

```
CALCulate:WLAN:CONSte:MARKer<x>:FREQuency <NRf>
CALCulate:WLAN:CONSte:MARKer<x>:FREQuency?
```

Arguments Floating point number that indicates the desired subcarrier to which the marker should be moved. Subcarriers are traditionally numbered with both positive and negative offsets from the CF for the modulation scheme. Valid integral values range from –maxsubcarrier to maxsubcarrier, where the actual values depend on the modulation standard. Specifying a value of <NRf> outside of the range of subcarriers present in the burst selects the nearest valid subcarrier. Subcarriers can also be specified by frequency (also determined by modulation standard).

Returns Returns a unitless number representing the subcarrier number.

Examples `CALC:WLAN:CONST:MARK2:FREQ 2.0000` sets the position of Marker 2 (M2) to Subcarrier 2 (assuming the units have been set to Subcarrier).

`CALC:WLAN:CONST:MARK2:FREQ?` might return `2.41575E+9`, indicating that Marker 2 (M2) is positioned at 2.41575 MHz.

CALCulate:WLAN:CONSte:MARKer<x>:MAGNitude? (Query Only)

Returns the magnitude value at the indicated marker position on the WLAN Constellation display.

<x> is optional. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:CONSte:MARKer<x>:MAGNitude?`

Returns Returns a unitless, floating point number indicating the magnitude at the selected marker position on the WLAN Constellation display.

Examples `CALC:WLAN:CONS:MARK1:MAGN?` might return `35` indicating the magnitude at Marker 1 (M1) is 35.

CALCulate:WLAN:CONSte:MARKer<x>:PHASe? (Query Only)

Returns the phase value at the indicated marker position on the WLAN Constellation display, in degrees.

<x>, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:CONSte:MARKer<x>:PHASe?`

Returns Phase value of the signal in degrees between -180 and +180 as an ASCII encoded floating point number in standard engineering notation.

Examples `CALC:WLAN:CONS:MARK1:PHAS?` might return 35.74, indicating the phase readout of Marker 1 (M1) is 35.74°.

CALCulate:WLAN:CONSte:MARKer<x>:TIME

Sets the time offset value at the selected marker position on the points trace on the WLAN Constellation display. The units are either Symbols or Seconds. To set/query the units, use the command [SENSe]:WLAN:UNIT:TIME.

Note that TIME is a measurement of neither absolute nor elapsed time, but symbol offset within the analyzed packet burst.

<x>, which is optional, is the desired marker. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Commanding a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax
```
CALCulate:WLAN:CONSte:MARKer<x>:TIME <NRf>
CALCulate:WLAN:CONSte:MARKer<x>:TIME?
```

Arguments Floating point number that indicates the 0 based symbol or time offset from the start of packet burst. Valid integral values range from 0 to the number of symbols in the burst. Specifying a value outside of the range of symbols present in the burst selects the nearest valid symbol.

Returns Returns a unitless integer value in floating point format that represents the symbol associated with the position of the indicated marker, or the time offset.

Examples `CALC:WLAN:CONST:MARK3:TIME -234.5` moves Marker 3 (M3) to the 362.42µS position on the trace (assuming the units have been set to Seconds).

`CALC:WLAN:CONST:MARK3:TIME?` might return 5.0000 indicating that Marker 3 is located at symbol 5 (assuming units have been set to Symbols).

CALCulate:WLAN:CONSte:MARKer<x>:TYPE? (Query Only)

Returns the data type of the subcarrier for data at the indicated marker position - either pilot or data.

<x>, which is optional, is the desired marker. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:CONSte:MARKer<x>:TYPE?`

Returns Returns the decoded value of the subcarrier type to which the marker is attached as an ASCII encoded floating point number in standard engineering notation. Values are:

1.000 if the marker type is a pilot marker.

2.000 if the marker type is a data marker.

Examples `CALC:WLAN:CONS:MARK2:TYPE?` might return `1.000` indicating Marker 2 (M2) is a pilot marker.

CALCulate:WLAN:CONSte:MARKer<x>:VALue? (Query Only)

Queries the value readout at the selected marker position on the points trace of the WLAN Constellation display.

<x>, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:CONSte:MARKer<x>:VALue?`

Returns Floating point number that represents the value readout at the selected marker position.

Examples `CALC:WLAN:CONS:MARK1:VAL?` might return 2.00, indicating the value readout at Marker 1 (M1) is 2.

CALCulate:WLAN:CRESponse:MARKer<x>:DELTa:X? (Query Only)

Queries the delta Subcarrier or frequency value (depending on which units have been set) at the selected marker position on the WLAN Channel Response display.

Use the command [SENSe]:WLAN:UNIT:FREQuency to set the units.

<x> is the desired marker to query. Marker 0 is not valid for the DELTa queries. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:CRESponse:MARKer<x>:DELTa:X?`

Returns Floating point number that represents the delta frequency value at the selected marker position.

Examples `CALC:WLAN:CRES:MARK2:DELT:X?` might return `1.28E+6` indicating that the delta frequency valuefor Marker 2 (M2) is 1.28 MHz.

CALCulate:WLAN:CRESponse:MARKer<x>:DELTa:Y? (Query Only)

Queries the vertical delta value for the points trace at the selected marker position on the WLAN Channel Response display, in dB or degree.

<x> is the desired marker to query. Marker 0 is not valid for the DELTa queries. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Marker 0 and the selected marker must be on the same plot (magnitude or phase); otherwise, it will result in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:CRESponse:MARKer<x>:DELTa:Y?`

Returns Returns the vertical delta value for the points trace in dB or degree at the selected marker position.

Examples `CALC:WLAN:CRES:MARK3:DELT:Y?` might return `-30`, indicating a vertical delta value of -30 dB at the Marker 3 (M3) position.

CALCulate:WLAN:CRESponse:MARKer<x>:MAXimum (No Query Form)

Moves the marker to the highest peak on the trace on the WLAN Channel Response display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:CRESponse:MARKer<x>:MAXimum`

Arguments None

Examples `CALC:WLAN:CRES:MARK2:MAX` moves the marker to the highest peak on the trace on the WLAN Channel Response display.

CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the marker to the next peak higher in amplitude on the WLAN Channel Response display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:HIGHer`

Arguments None

Examples `CALC:WLAN:CRES:MARK2:PEAK:HIGHer` moves Marker 2 (M2) to the next peak higher in amplitude on the trace.

CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the marker to the next peak on the left on the trace in the WLAN Channel Response display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:LEFT`

Arguments None

Examples `CALC:WLAN:CRES:MARK2:PEAK:LEFT` moves Marker 2 (M2) to the next peak on the left on the trace in the WLAN Channel Response display.

CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the marker to the next peak lower in amplitude on the trace on the WLAN Channel Response display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:LOWer`

Arguments None

Examples `CALC:WLAN:CRES:MARK2:PEAK:LOW` moves Marker 2 (M2) to the next peak lower on the trace.

CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the marker to the next peak to the right on the trace on the WLAN Channel Response display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:CRESponse:MARKer<x>:PEAK:RIGHt`

Arguments None

Examples `CALC:WLAN:CRES:MARK2:PEAK:RIGH` moves Marker 2 (M2) to the next peak to the next peak on the right on the trace.

CALCulate:WLAN:CRESponse:MARKer<x>:X

Sets or queries the value of the horizontal position (units in Frequency or Subcarrier) for the selected marker on the points trace in the WLAN Channel Response display.

To set the units, use the command [SENSe]:WLAN:UNIT:FREQuency.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax

```
CALCulate:WLAN:CRESponse:MARKer<x>:X <NRf>
CALCulate:WLAN:CRESponse:MARKer<x>:X?
```

Arguments Floating point number that specifies the value of the horizontal position of the marker. Range: Start to Stop frequency (left to right edge of the horizontal axis). Using an out-of-range value causes an execution error (-222, “Data out of range”).

Examples `CALC:WLAN:CRES:MARK3:X 22` places Marker 3 at 22 Subcarrier on the trace (assuming the units have been set to Subcarrier).

CALCulate:WLAN:CRESponse:MARKer<x>:Y? (Query Only)

Returns the value of the amplitude at the selected marker position in the WLAN Channel Response measurement.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax

```
CALCulate:WLAN:CRESponse:MARKer<x>:Y?
```

Returns Floating point number that indicates the amplitude value at the selected marker position in dB or degrees.

Examples `CALC:WLAN:CRES:MARK1:Y?` might return `33.393856E-3` indicating the amplitude of Marker 1 is 0.033 dB.

CALCulate:WLAN:EVM:MARKer<x>:AVERage:SUBCarrier? (Query Only)

Queries the EVM value at the selected marker position on the average trace on the WLAN EVM display subcarrier graph.

To query the marker value of the points trace, use the query.

<x>, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:EVM:MARKer<x>:AVERage:SUBCarrier?`

Returns Returns a floating point number that represents the EVM value at the selected marker position on the average trace on the WLAN EVM display subcarrier graph.

Examples `CALC:WLAN:EVM:MARK3:AVER:SUBC?` might return `16.4`, indicating an EVM value of 16.4 percent at the Marker 3 (M3) position on the average trace on the WLAN EVM display subcarrier graph.

CALCulate:WLAN:EVM:MARKer<x>:AVERage:SYMBol? (Query Only)

Queries the EVM value at the selected marker position on the average trace on the WLAN EVM display symbol graph.

To query the marker value of the points trace, use the query.

<x>, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:EVM:MARKer<x>:AVERage:SYMBol?`

Returns Returns a floating point number that represents the EVM value at the selected marker position on the average trace on the WLAN EVM display symbol graph.

Examples `CALC:WLAN:EVM:MARK3:AVER:SYMB?` might return `14.5`, indicating an EVM value of 14.5 percent at the Marker 3 (M3) position on the average trace on the WLAN EVM display symbol graph.

CALCulate:WLAN:EVM:MARKer<x>:DELTa:SUBCarrier? (Query Only)

Queries the delta Subcarrier or frequency value (depending on which units have been set) at the selected marker position on the WLAN EVM display.

Use the command [SENSe]:WLAN:UNIT:FREQuency to set the units.

<x>, which is optional, is the desired marker to query. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Marker 0 is not valid for the DELTa queries. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:EVM:MARKer<x>:DELTa:SUBCarrier?`

Returns Returns a floating point number that represents the delta subcarrier or frequency value at the selected marker position on the WLAN EVM display.

Examples `CALC:WLAN:EVM:MARK3:DELTa:SUBCarrier?` might return `9.375 MHz`, indicating a delta frequency value of 9.375 MHz at the Marker 3 (M3) position.

CALCulate:WLAN:EVM:MARKer<x>:DELTa:SYMBol? (Query Only)

Queries the delta time value in seconds or symbols (depending on which units have been set) at the selected marker position on the WLAN EVM display.

Use the command [SENSe]:WLAN:UNIT:TIME to set the units.

<x>, which is optional, is the desired marker to query. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Marker 0 is not valid for the DELTa queries. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:EVM:MARKer<x>:DELTa:SYMBol?`

Returns Returns the delta time value at the selected marker position on the WLAN EVM display.

Examples `CALC:WLAN:EVM:MARK3:DELTa:SYMB?` might return `4.0E-6 µs`, indicating a delta time value of 4 µs at the Marker 3 (M3) position (assuming the units have been set to Seconds).

CALCulate:WLAN:EVM:MARKer<x>:DELTa:Y? (Query Only)

Queries the vertical delta value for the points trace at the selected marker position on the WLAN EVM display.

<x>, which is optional, is the desired marker to query. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Marker 0 is not valid for the DELTa queries. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:EVM:MARKer<x>:DELTa:Y?`

Returns Returns the vertical delta value for the points trace at the selected marker position.

Examples `CALC:WLAN:EVM:MARK3:DELTa:Y?` might return `-212.91`, indicating a vertical delta value of -212.91 percent at the Marker 3 (M3) position.

CALCulate:WLAN:EVM:MARKer<x>:FREQuency

Sets or queries the frequency value of the marker on the WLAN EVM display. The units are either Frequency in Hz or Subcarrier number. To set the units, use the command [SENSe]:WLAN:UNIT:FREQuency.

Note that FREQuency is not an actual measurement of frequency, but is the subcarrier offset within the analyzed packet burst.

<x>, which is optional, is the desired marker to use. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Commanding a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax
```
CALCulate:WLAN:EVM:MARKer<x>:FREQuency <NRf>
CALCulate:WLAN:EVM:MARKer<x>:FREQuency?
```

Arguments Floating point number that indicates the desired subcarrier or frequency to which the marker should be moved. Subcarriers are traditionally numbered with both positive and negative offsets from the CF for the modulation scheme. Valid integral values range from –maxsubcarrier to maxsubcarrier, where the actual values depend on the modulation standard. Specifying a value of <NRf> outside of the range of subcarriers present in the burst selects the nearest valid subcarrier. Subcarriers can also be specified by frequency (also determined by modulation standard).

Returns Returns a unitless number representing the subcarrier number or frequency in Hz.

Examples `CALC:WLAN:EVM:MARK2:FREQ 2.0000` sets the position of Marker 2 (M2) to Subcarrier 2 (assuming the units have been set to Subcarrier).

`CALC:WLAN:EVM:MARK2:FREQ?` might return `2.41575E+9`, indicating that Marker 2 (M2) is positioned at 2.41575 MHz.

CALCulate:WLAN:EVM:MARKer<x>:TIME

Sets the vertical marker value of the points trace on the WLAN EVM display. The units are either Symbols or Seconds. To set/query the units, use the command [SENSe]:WLAN:UNIT:TIME.

Note that TIME is a measurement of neither absolute nor elapsed time, but symbol offset within the analyzed packet burst.

<x>, which is optional, is the desired marker to use. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Commanding a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax
```
CALCulate:WLAN:EVM:MARKer<x>:TIME <NRf>
CALCulate:WLAN:EVM:MARKer<x>:TIME?
```

Arguments Floating point number that indicates the 0 based symbol or time offset from the start of packet burst. Valid integral values range from 0 to the number of symbols in the burst. Specifying a value outside of the range of symbols present in the burst selects the nearest valid symbol.

Returns Returns a unitless integer value in floating point format that represents the symbol associated with the position of the indicated marker, or the time offset.

Examples `CALC:WLAN:EVM:MARK3:TIME 362.42E-6` moves Marker 3 (M3) to the 362.42µS position on the trace (assuming the units have been set to Seconds).

CALC:WLAN:EVM:MARK3:TIME? might return 5.0000 indicating that Marker 3 is located at symbol 5 (assuming units have been set to Symbols).

CALCulate:WLAN:EVM:MARKer<x>:VALue? (Query Only)

Queries the EVM value at the selected marker position on the points trace of the WLAN EVM display.

The EVM value of a marker on the average trace is queried by using the and CALCulate:WLAN:EVM:MARKer<x>:AVERage:SYMBol? queries.

<x>, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax CALCulate:WLAN:EVM:MARKer<x>:VALue?

Returns Returns the EVM value at the selected marker position on the WLAN EVM display.

Examples CALC:WLAN:EVM:MARK3:VAL? might return 11.9, indicating an EVM value of 11.9 percent for Marker 3 (M3).

CALCulate:WLAN:FLATness:MARKer<x>:MAXimum (No Query Form)

Positions the specified marker at the maximum point on the trace in the WLAN Spectral Flatness display.

NOTE. *WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:FLATness:MARKer<x>:MAXimum`

Arguments None

Examples `CALC:WLAN:FLAT:MARK2:MAX` positions the Marker 2 (M2) at the maximum point on the trace.

CALCulate:WLAN:FLATness:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the specified marker to the next peak higher in amplitude on the trace on the WLAN Spectral Flatness display.

NOTE. *WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:FLATness:MARKer<x>:PEAK:HIGHer`

Arguments None

Examples `CALC:WLAN:FLAT:MARK2:PEAK:HIGH` moves Marker 2 (M2) to the next peak higher in amplitude on the trace.

CALCulate:WLAN:FLATness:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the specified marker to the next peak to the left on the trace in the WLAN Spectral Flatness display.

NOTE. *WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:FLATness:MARKer<x>:PEAK:LEFT`

Arguments None

Examples `CALC:WLAN:FLAT:MARK2:PEAK:LEFT` moves Marker 2 (M2) to the next peak to the left on the trace .

CALCulate:WLAN:FLATness:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the marker to the next peak lower in amplitude on the trace on the WLAN Spectral Flatness display.

NOTE. *WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:FLATness:MARKer<x>:PEAK:LOWer`

Arguments None

Examples `CALC:WLAN:FLAT:MARK2:PEAK:LOW` moves Marker 2 (M2) to the next peak lower on the trace.

CALCulate:WLAN:FLATness:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the marker to the next peak to the right on the trace on the WLAN Spectral Flatness display.

NOTE. *WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:FLATness:MARKer<x>:PEAK:RIGHt`

Arguments None

Examples `CALC:WLAN:FLAT:MARK2:PEAK:RIGH` moves Marker 2 (M2) to the next peak to the next peak on the right on the trace.

CALCulate:WLAN:FLATness:MARKer<x>:X

Sets or queries the horizontal position (units in Frequency or Subcarrier) for the selected marker on the points trace in the WLAN Spectral Flatness display.

To set the units, use the command [SENSe]:WLAN:UNIT:FREQuency.

NOTE. *WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax
```
CALCulate:WLAN:FLATness:MARKer<x>:X <NRf>
CALCulate:WLAN:FLATness:MARKer<x>:X?
```

Arguments Floating point number that specifies the horizontal position of the marker. Range: Start to Stop frequency (left to right edge of the horizontal axis). Using an out-of-range value causes an execution error (-222, “Data out of range”).

Examples `CALC:WLAN:FLAT:MARK3:X 22` places Marker 3 at 22 Subcarrier on the trace (assuming the units have been set to Subcarrier).

CALCulate:WLAN:FLATness:MARKer<x>:Y? (Query Only)

Returns the value of the amplitude (vertical position) at the selected marker position in the WLAN Spectral Flatness measurement.

NOTE. *WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:FLATness:MARKer<x>:Y?`

Returns Floating point number that indicates the amplitude at the selected marker position in dB.

Examples `CALC:WLAN:FLAT:MARK1:Y?` might return `33.393856E-3` indicating the amplitude of Marker 1 is 0.033 dB.

CALCulate:WLAN:MERRor:MARKer<x>:AVERage:SUBCarrier? (Query Only)

Queries the value readout at the selected marker position on the average trace on the WLAN Magnitude Error display.

To query the value readout on the symbol graph, use CALCulate:WLAN:MERRor:MARKer<x>:AVERage:SYMBol?

<x>, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:MERRor:MARKer<x>:AVERage:SUBCarrier?`

Returns Returns a floating point number that represents the value readout at the selected marker position on the average trace on the WLAN Magnitude Error display subcarrier graph.

Examples `CALC:WLAN:MERR:MARK2:AVERAGE:SUBC?` might return `16.4`, indicating a value readout of 16.4 percent at the Marker 3 (M3) position on the average trace on the WLAN Magnitude Error display subcarrier graph.

CALCulate:WLAN:MERRor:MARKer<x>:AVERage:SYMBol? (Query Only)

Queries the value readout at the selected marker position on the average trace on the WLAN Magnitude Error display.

To query the value readout on the subcarrier graph, use CALCulate:WLAN:MERRor:MARKer<x>:AVERage:SUBCarrier?

<x>, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:MERRor:MARKer<x>:AVERage:SYMBol?`

Returns Returns a floating point number that represents the value readout at the selected marker position on the average trace on the WLAN Magnitude Error display symbol graph.

Examples `CALC:WLAN:MERR:MARK2:AVERAGE:SUBC?` might return `14.5`, indicating a Magnitude Error value of 14.5 percent at the Marker 3 (M3) position on the average trace on the WLAN Magnitude Error display.

CALCulate:WLAN:MERRor:MARKer<x>:DELTa:SUBCarrier? (Query Only)

Queries the delta Subcarrier or frequency value (depending on which units have been set) at the selected marker position on the WLAN Magnitude Error display.

Use the command [SENSe]:WLAN:UNIT:FREQuency to set the units.

<x>, which is optional, is the desired marker to query. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Marker 0 is not valid for the DELTa queries. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:MERRor:MARKer<x>:DELTa:SUBCarrier?`

Returns Returns a floating point number that represents the delta subcarrier or frequency value at the selected marker position on the WLAN Magnitude Error display.

Examples `CAL:WLAN:MERR:MARK2:DELTA:SUBC?` might return `9.375 MHz`, indicating a delta frequency value of 9.375 MHz at the Marker 3 (M3) position.

CALCulate:WLAN:MERRor:MARKer<x>:DELTa:SYMBol? (Query Only)

Queries the delta time value in seconds or symbols (depending on which units have been set) at the selected marker position on the WLAN Magnitude Error display.

Use the command [SENSe]:WLAN:UNIT:TIME to set the units.

<x>, which is optional, is the desired marker to query. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Marker 0 is not valid for the DELTa queries. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:MERRor:MARKer<x>:DELTa:SYMBol?`

Returns Returns the delta time value at the selected marker position on the WLAN Magnitude Error display.

Examples `CALC:WLAN:MERR:MARK3:DELTa:SYMB?` might return `4.0E-6 µs`, indicating a delta time value of 4 µs at the Marker 3 (M3) position (assuming the units have been set to Seconds).

CALCulate:WLAN:MERRor:MARKer<x>:DELTa:Y? (Query Only)

Queries the vertical delta value for the points trace at the selected marker position on the WLAN Magnitude Error display.

<x>, which is optional, is the desired marker to query. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Marker 0 is not valid for the DELTa queries. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:MERRor:MARKer<x>:DELTa:Y?`

Returns Returns the vertical delta value for the points trace at the selected marker position.

Examples `CALC:WLAN:MERR:MARK3:DELTa:Y?` might return `-212.91`, indicating a vertical delta value of -212.91 percent at the Marker 3 (M3) position.

CALCulate:WLAN:MERRor:MARKer<x>:FREQuency

Sets or queries the frequency value of the marker on the WLAN Magnitude Error display. The units are either Frequency in Hz or Subcarrier number. To set the units, use the command [SENSe]:WLAN:UNIT:FREQuency.

Note that FREQuency is not an actual measurement of frequency, but is the subcarrier offset within the analyzed packet burst.

<x>, which is optional, is the desired marker to use. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Commanding a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax

```
CALCulate:WLAN:MERRor:MARKer<x>:FREQuency <NRf>
CALCulate:WLAN:MERRor:MARKer<x>:FREQuency?
```

Arguments Floating point number that indicates the desired subcarrier or frequency to which the marker should be moved. Subcarriers are traditionally numbered with both positive and negative offsets from the CF for the modulation scheme. Valid integral values range from –maxsubcarrier to maxsubcarrier, where the actual values depend on the modulation standard. Specifying a value of <NRf> outside of the range of subcarriers present in the burst selects the nearest valid subcarrier. Subcarriers can also be specified by frequency (also determined by modulation standard).

Returns Returns a unitless number representing the subcarrier number or frequency value in Hz.

Examples `CALC:WLAN:MERR:MARK2:FREQ 2.0000` sets the position of Marker 2 (M2) to Subcarrier 2 (assuming the units have been set to Subcarrier).

`CALC:WLAN:MERR:MARK2:FREQ?` might return `2.41575E+9`, indicating that Marker 2 (M2) is positioned at 2.41575 MHz.

CALCulate:WLAN:MERRor:MARKer<x>:TIME

Sets the time offset value at the selected marker position on the points trace on the WLAN Magnitude Error display. The units are either Symbols or Seconds. To set/query the units, use the command [SENSe]:WLAN:UNIT:TIME.

Note that TIME is a measurement of neither absolute nor elapsed time, but symbol offset within the analyzed packet burst.

<x>, which is optional, is the desired marker. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Commanding a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax

```
CALCulate:WLAN:MERRor:MARKer<x>:TIME <NRf>
CALCulate:WLAN:MERRor:MARKer<x>:TIME?
```

Arguments Floating point number that indicates the 0 based symbol or time offset from the start of packet burst. Valid integral values range from 0 to the number of symbols in the burst. Specifying a value outside of the range of symbols present in the burst selects the nearest valid symbol.

Returns Returns a unitless integer value in floating point format that represents the time associated with the position of the indicated marker, or the time offset.

Examples `CALC:WLAN:MERR:MARK3:TIME 362.42E-6` moves Marker 3 (M3) to the 362.42 µS position on the trace (assuming the units have been set to Seconds).

`CALC:WLAN:MERR:MARK3:TIME?` might return 5.0000 indicating that Marker 3 is located at symbol 5 (assuming units have been set to Symbols).

CALCulate:WLAN:MERRor:MARKer<x>:VALue? (Query Only)

Queries the value readout at the selected marker position on the points trace of the WLAN Magnitude Error display.

The value readout of a marker on the average trace is queried by using the and CALCulate:WLAN:MERRor:MARKer<x>:AVERage:SYMBol? queries.

<x>, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:MERRor:MARKer<x>:VALue?`

Returns Returns the decoded symbol value at the selected marker position on the WLAN Magnitude Error display.

Examples `CALC:WLAN:MERR:MARK3:VAL?` might return `11.9`, indicating a value readout of 11.9 for Marker 3 (M3).

CALCulate:WLAN:PERRor:MARKer<x>:AVERage:SUBCarrier? (Query Only)

Queries the Phase Error value at the selected marker position on the average trace on the WLAN Phase Error display subcarrier graph.

To query the marker value of the points trace, use the query.

<x>, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:PERRor:MARKer<x>:AVERage:SUBCarrier?`

Returns Returns a floating point number that represents the Phase Error value at the selected marker position on the average trace on the WLAN Phase Error display subcarrier graph.

Examples CALC:WLAN:PERR:MARK2:AVERAGE:SUBC? might return 16.4, indicating a Phase Error value of 16.4 percent at the Marker 3 (M3) position on the average trace on the WLAN Phase Error display subcarrier graph.

CALCulate:WLAN:PERRor:MARKer<x>:AVERage:SYMBol? (Query Only)

Queries the Phase Error value at the selected marker position on the average trace on the WLAN Phase Error display symbol graph.

To query the marker value of the points trace, use the query.

<x>, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Group Calculate commands

Syntax CALCulate:WLAN:PERRor:MARKer<x>:AVERage:SYMBol?

Returns Returns a floating point number that represents the Phase Error value at the selected marker position on the average trace on the WLAN Phase Error display symbol graph.

Examples CALC:WLAN:PERR:MARK2:AVERAGE:SUBC? might return 14.5, indicating a Phase Error value of 14.5 percent at the Marker 3 (M3) position on the average trace on the WLAN Phase Error display symbol graph.

CALCulate:WLAN:PERRor:MARKer<x>:DELTa:SUBCarrier? (Query Only)

Queries the delta Subcarrier or frequency value (depending on which units have been set) at the selected marker position on the WLAN Phase Error display.

Use the command [SENSe]:WLAN:UNIT:FREQuency to set the units.

<x>, which is optional, is the desired marker to query. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Marker 0 is not valid for the DELTa queries. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:PERRor:MARKer<x>:DELTa:SUBCarrier?`

Returns Returns a floating point number that represents the delta subcarrier or frequency value at the selected marker position on the WLAN Phase Error display.

Examples `CAL:WLAN:PERR:MARK2:DELTA:SUBC?` might return `9.375 MHz`, indicating a delta frequency value of 9.375 MHz at the Marker 3 (M3) position.

CALCulate:WLAN:PERRor:MARKer<x>:DELTa:SYMBol? (Query Only)

Queries the delta time value in seconds or symbols (depending on which units have been set) at the selected marker position on the WLAN Phase Error display.

Use the command [SENSe]:WLAN:UNIT:TIME to set the units.

<x>, which is optional, is the desired marker to query. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Marker 0 is not valid for the DELTa queries. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:PERRor:MARKer<x>:DELTa:SYMBol?`

Returns Returns the delta time value at the selected marker position on the WLAN Phase Error display.

Examples `CALC:WLAN:PERR:MARK3:DELTa:SYMB?` might return `4.0E-6 µs`, indicating a delta time value of 4 µs at the Marker 3 (M3) position (assuming the units have been set to Seconds).

CALCulate:WLAN:PERRor:MARKer<x>:DELTa:Y? (Query Only)

Queries the vertical delta value for the points trace at the selected marker position on the WLAN Phase Error display.

<x>, which is optional, is the desired marker to query. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Marker 0 is not valid for the DELTa queries. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:PERRor:MARKer<x>:DELTa:Y?`

Returns Returns the vertical delta value for the points trace at the selected marker position.

Examples `CALC:WLAN:PERR:MARK3:DELTa:Y?` might return `-212.91`, indicating a vertical delta value of -212.91 percent at the Marker 3 (M3) position.

CALCulate:WLAN:PERRor:MARKer<x>:FREQuency

Sets or queries the frequency value at the selected marker position on the WLAN Phase Error display. The units are either Frequency in Hz or Subcarrier number. To set the units, use the command [SENSe]:WLAN:UNIT:FREQuency.

Note that FREQuency is not an actual measurement of frequency, but is the subcarrier offset within the analyzed packet burst.

<x>, which is optional, is the desired marker to use. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Commanding a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax
```
CALCulate:WLAN:PERRor:MARKer<x>:FREQuency <NRf>
CALCulate:WLAN:PERRor:MARKer<x>:FREQuency?
```

Arguments Floating point number that indicates the desired subcarrier or frequency to which the marker should be moved. Subcarriers are traditionally numbered with both positive and negative offsets from the CF for the modulation scheme. Valid integral values range from –maxsubcarrier to maxsubcarrier, where the actual values depend on the modulation standard. Specifying a value of <NRf> outside of the range of subcarriers present in the burst selects the nearest valid subcarrier. Subcarriers can also be specified by frequency (also determined by modulation standard).

Returns Returns a unitless number representing the subcarrier number or frequency value in Hz.

Examples `CALC:WLAN:PERR:MARK2:FREQ 2.0000` sets the position of Marker 2 (M2) to Subcarrier 2 (assuming the units have been set to Subcarrier).

`CALC:WLAN:PERR:MARK2:FREQ?` might return `2.41575E+9`, indicating that Marker 2 (M2) is positioned at 2.41575 MHz.

CALCulate:WLAN:PERRor:MARKer<x>:TIME

Sets the time value (offset) at the selected marker position on the points trace on the WLAN Phase Error display. The units are either Symbols or Seconds. To set/query the units, use the command [SENSe]:WLAN:UNIT:TIME.

Note that TIME is a measurement of neither absolute nor elapsed time, but symbol offset within the analyzed packet burst.

<x>, which is optional, is the desired marker to use. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Commanding a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax

```
CALCulate:WLAN:PERRor:MARKer<x>:TIME <NRf>
CALCulate:WLAN:PERRor:MARKer<x>:TIME?
```

Arguments

Floating point number that indicates the 0 based symbol or time offset from the start of packet burst. Valid integral values range from 0 to the number of symbols in the burst. Specifying a value outside of the range of symbols present in the burst selects the nearest valid symbol.

Returns

Returns a unitless integer value in floating point format that represents the symbol associated with the position of the indicated marker, or the time offset.

Examples

`CALC:WLAN:PERR:MARK3:TIME 362.42E-6` moves Marker 3 (M3) to the 362.42µS position on the trace (assuming the units have been set to Seconds).

`CALC:WLAN:PERR:MARK3:TIME?` might return 5.0000 indicating that Marker 3 is located at symbol 5 (assuming units have been set to Symbols).

CALCulate:WLAN:PERRor:MARKer<x>:VALue? (Query Only)

Queries the Phase Error value at the selected marker position on the points trace of the WLAN Phase Error display.

The Phase Error value of a marker on the average trace is queried by using the and CALCulate:WLAN:PERRor:MARKer<x>:AVERage:SYMBol? queries.

<x>, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Calculate commands

Syntax

```
CALCulate:WLAN:PERRor:MARKer<x>:VALue?
```

Returns

Returns the Phase Error value at the selected marker position on the WLAN Phase Error display.

Examples `CALC:WLAN:PERR:MARK3:VAL?` might return `11.9`, indicating a Phase Error value of 11.9 percent for Marker 3 (M3).

CALCulate:WLAN:PVTime:MARKer<x>:DELTa:X? (Query Only)

Returns the delta time value at the selected marker position in the WLAN Power vs. Time display.

<x> is the desired marker to query. Marker 0 is not valid for the DELTa queries. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:PVTime:MARKer<x>:DELTa:X?`

Returns Floating point number that represents the delta time value at the selected marker position.

Examples `CALC:WLAN:PVT:MARK2:DELT:X?` might return `1.28E-6` indicating that the delta time value for Marker 2 (M2) is 1.28 µs.

CALCulate:WLAN:PVTime:MARKer<x>:DELTa:Y? (Query Only)

Queries the vertical delta value for the points trace at the selected marker position on the WLAN Power vs. Time display, using the currently selected Amplitude unit. Set/query the amplitude unit with the command `SENSe:POWer:UNITs`.

<x> is the desired marker to query. Marker 0 is not valid for the DELTa queries. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. Valid integral values range from 1 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:PVTime:MARKer<x>:DELTa:Y?`

Returns Returns the vertical delta value for the points trace in the current Amplitude units at the selected marker position.

Examples `CALC:WLAN:PVT:MARK3:DELT:Y?` might return `-12.91`, indicating a vertical delta value of -12.91 dB at the Marker 3 (M3) position.

CALCulate:WLAN:PVTime:MARKer<x>:MAXimum (No Query Form)

Positions the specified marker at the maximum point on the trace in the WLAN Power vs. Time display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:PVTime:MARKer<x>:MAXimum`

Arguments None

Examples `CALC:WLAN:PVT:MARK2:MAX` positions the Marker 2 (M2) at the maximum point on the trace.

CALCulate:WLAN:PVTime:MARKer<x>:PEAK:HIGHer (No Query Form)

Moves the specified marker to the next peak higher in amplitude on the trace on the WLAN Power vs. Time display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax CALCulate:WLAN:PVTime:MARKer<x>:PEAK:HIGHer

Arguments None

Examples CALC:WLAN:PVT:MARK2:PEAK:HIGHer moves Marker 2 (M2) to the next peak higher in amplitude on the trace.

CALCulate:WLAN:PVTime:MARKer<x>:PEAK:LEFT (No Query Form)

Moves the specified marker to the next peak on the left on the trace in the WLAN Power vs. Time display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax CALCulate:WLAN:PVTime:MARKer<x>:PEAK:LEFT

Arguments None

Examples CALC:WLAN:PVT:MARK2:PEAK:LEFT moves Marker 2 (M2) to the next peak on the left on the trace in the WLAN Power vs. Time display.

CALCulate:WLAN:PVTime:MARKer<x>:PEAK:LOWer (No Query Form)

Moves the marker to the next peak lower in amplitude on the trace on the WLAN Power vs. Time display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax CALCulate:WLAN:PVTime:MARKer<x>:PEAK:LOWer

Arguments None

Examples `CALC:WLAN:PVT:MARK2:PEAK:LOW` moves Marker 2 (M2) to the next peak lower on the trace.

CALCulate:WLAN:PVTime:MARKer<x>:PEAK:RIGHt (No Query Form)

Moves the marker to the next peak to the right on the trace on the WLAN Power vs. Time display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:PVTime:MARKer<x>:PEAK:RIGHt`

Arguments None

Examples `CALC:WLAN:PVT:MARK2:PEAK:RIGH` moves Marker 2 (M2) to the next peak on the right on the trace.

CALCulate:WLAN:PVTime:MARKer<x>:X

Sets or queries the value of the horizontal position (in seconds) for the selected marker on the points trace in the WLAN Power vs. Time display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax
```
CALCulate:WLAN:PVTime:MARKer<x>:X <NRf>
CALCulate:WLAN:PVTime:MARKer<x>:X?
```

Arguments Floating point number that specifies the value of the horizontal position of the marker. Range: Start to Stop time (left to right edge of the horizontal axis). Using an out-of-range value causes an execution error (-222, "Data out of range").

Examples `CALC:WLAN:PVT:MARK3:X 1E-6` places Marker 3 at 1 µs on the trace.

CALCulate:WLAN:PVTime:MARKer<x>:Y? (Query Only)

Returns the amplitude (vertical position) at the selected marker position in the WLAN Power vs. Time display, using the currently selected Amplitude units. You can set/query the Amplitude units with the command `SENSe:POWer:UNITs`.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:PVTime:MARKer<x>:Y?`

Returns Floating point number that indicates the amplitude at the selected marker position in the currently selected Amplitude units.

Examples `CALC:WLAN:PVT:MARK1:Y?` might return `33.393856E-3` indicating the amplitude at the Marker 1 position is 0.033 dBm.

CALCulate:WLAN:TABLe:MARKer<x>:FREQuency

Assigns or queries the frequency value of the marker in the WLAN Symbol Table. The units are either Frequency in Hz or Subcarrier number. To set the units, use the command [SENSe]:WLAN:UNIT:FREQuency.

Note that FREQuency is not an actual measurement of frequency, but is the subcarrier offset within the analyzed packet burst.

<x>, which is optional, is the desired marker to assign. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax
```
CALCulate:WLAN:TABLe:MARKer<x>:FREQuency <NRf>
CALCulate:WLAN:TABLe:MARKer<x>:FREQuency?
```

Arguments Floating point number that indicates the desired subcarrier or frequency to which the marker should be assigned. Subcarriers are traditionally numbered with both positive and negative offsets from the CF for the modulation scheme. Valid integral values range from –maxsubcarrier to maxsubcarrier, where the actual values depend on the modulation standard. Specifying a value of <NRf> outside of the range of subcarriers present in the burst selects the nearest valid subcarrier. Subcarriers can also be specified by frequency (also determined by modulation standard).

Returns Returns a unitless number representing the subcarrier number or frequency in Hz.

Examples `CALC:WLAN:TABLE:MARK3:FREQ 2.0000` assigns Subcarrier 2 to Marker 3 (M3) (assuming the units have been set to Subcarrier).

`CALC:WLAN:TABLEM:MARK2:FREQ?` might return `2.00000E+9`, indicating that Marker 2 (M2) is assigned 2.00 MHz.

CALCulate:WLAN:TABLe:MARKer<x>:TIME

Sets or queries the vertical value at the selected marker position on the WLAN Symbol Table. The units are either Symbols or Seconds. To set/query the units, use the command [SENSe]:WLAN:UNIT:TIME.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax
```
CALCulate:WLAN:TABLe:MARKer<x>:TIME <NRf>
CALCulate:WLAN:TABLe:MARKer<x>:TIME?
```

Arguments Floating point number that indicates the 0 based symbol or time offset from the start of packet burst. Valid integral values range from 0 to the number of symbols in the burst. Specifying a value outside of the range of symbols present in the burst selects the nearest valid symbol.

Returns Returns a unitless integer value in floating point format that represents the symbol associated with the indicated marker, or the time offset.

Examples `CALC:WLAN:TABLE:MARK3:TIME 362.42E-6` assigns 362.42 µs to Marker 3 (M3) (assuming the units have been set to Seconds).

`CALC:WLAN:TABLE:MARK3:TIME?` might return 5.0000 indicating that symbol 5 is assigned to Marker 3.

CALCulate:WLAN:TABLe:MARKer<x>:VALue? (Query Only)

Queries the decoded symbol value at the selected marker position on the WLAN Symbol Table display.

<x>, which is optional, is the desired marker to query. 0 indicates MR, the reference marker. 1, 2, 3, etc. indicate markers M1, M2, M3, etc. If <x> is not supplied, the command works on the currently selected marker. Valid integral values range from 0 to the number of markers active for the display. Querying a marker which does not exist results in an execution error.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Calculate commands

Syntax `CALCulate:WLAN:TABLe:MARKer<x>:VALue?`

Returns Returns the decoded symbol value at the selected marker position on the WLAN Symbol Table display.

Examples `CALC:WLAN:MERR:MARK3:VAL?` might return `11.9`, indicating a decoded symbol value of 11.9 for Marker 3 (M3).

CALibration:ABORt (No Query Form)

This command does not apply to SignalVu-PC or connected instruments.

Aborts any actions related to the alignments in progress.

Conditions Measurement views: All

Group Calibration commands

Syntax
```
CALibration:ABORt
```

Arguments None

Examples `CALIBRATION:ABORT` aborts any actions related to the alignments in progress.

CALibration:AUTO

This command does not apply to SignalVu-PC or connected instruments.

Sets or queries whether or not to run alignments automatically.

Conditions Measurement views: All

Group Calibration commands

Syntax
```
CALibration:AUTO { OFF | ON | 0 | 1 }
CALibration:AUTO?
```

Arguments

`OFF` or `0` runs alignments on user request.
Use the *CAL command to perform alignments.

`ON` or `1` runs alignments as needed without user intervention.
You have to restart measurement if interrupted.

Examples `CALIBRATION:AUTOON` runs alignments automatically as needed.

CALibration:CORRection:EXTernal:EDIT<x>:LABel

This command does not apply to SignalVu-PC or connected instruments.

Sets or queries the name of the external loss table.

The parameter <x> = 1 to 3 represent the External Loss Table 1 to 3, respectively.

Conditions Measurement views: All

Group Calibration commands

Syntax
```
CALibration:CORRection:EXTernal:EDIT<x>:LABel <name>
CALibration:CORRection:EXTernal:EDIT<x>:LABel?
```

Arguments `<name>::=<string>` specifies the name of the external loss table.

Examples `CALIBRATION:CORRECTION:EXTERNAL:EDIT1:LABEL "Sample Table 1”` names the External Loss Table 1 "Sample Table 1".

CALibration:CORRection:EXTernal:EDIT<x>:NEW (No Query Form)

This command does not apply to SignalVu-PC or connected instruments.

Creates a new external loss table.

The parameter <x> = 1 to 3 represent the External Loss Table 1 to 3, respectively.

Conditions Measurement views: All

Group Calibration commands

Syntax
```
CALibration:CORRection:EXTernal:EDIT<x>:NEW
<freq(1)>,<loss(1)>,<freq(2)>,<loss(2)>,...,<freq(n)>,<loss(n)>
```

Arguments `<freq(n)>,<loss(n)>` specifies a pair of frequency (<NR3> in Hz) and loss (<NR3> in dB) in the external loss table.

The setting range is:

- RSA5103B – 0 to 3 GHz
- RSA5115B – 0 to 15 GHz
- RSA5126B – 0 to 26.5 GHz

Loss: -50 to +80 dB. (A negative value means a gain.)

Examples `CALIBRATION:CORRECTION:EXTERNAL:EDIT1:NEW 1.0E+9,2.2, 1.5E+9,2.3` creates the External Loss Table 1 specifying the loss of 2.2 dB and 2.3 dB at the frequency of 1 GHz and 1.5 GHz, respectively.

CALibration:CORRection:EXTernal:EDIT<x>:STATe

This command does not apply to SignalVu-PC or connected instruments.

Determines whether to enable or disable the external loss table.

The parameter <x> = 1 to 3 represent the External Loss Table 1 to 3, respectively.

Conditions Measurement views: All

Group Calibration commands

Syntax
```
CALibration:CORRection:EXTernal:EDIT<x>:STATe { OFF | ON | 0 | 1 }
CALibration:CORRection:EXTernal:EDIT<x>:STATe?
```

Arguments `OFF` or `0` disables the external loss table.

`ON` or `1` enables the external loss table.

NOTE. *You can enable one or more tables at the same time.*

Examples `CALIBRATION:CORRECTION:EXTERNAL:EDIT3:STATE ON` enables the External Loss Table 3.

CALibration:CORRection:EXTernal:GAIN[:MAGNitude]

This command does not apply to SignalVu-PC or connected instruments.

Sets or queries the external gain value. It can be enabled or disabled using the CALibration:CORRection:EXTernal:GAIN:STATe command.

Conditions Measurement views: All

Group Calibration commands

Syntax
```
CALibration:CORRection:EXTernal:GAIN[:MAGNitude] <value>
CALibration:CORRection:EXTernal:GAIN[:MAGNitude]?
```

Arguments <value>::=<NRf> specifies the external gain value. Range: -50 to +30 dB.

Examples CALIBRATION:CORRECTION:EXTERNAL:GAIN:MAGNITUDE -10 sets the external gain to -10 dB.

CALibration:CORRection:EXTernal:GAIN:STATe

This command does not apply to SignalVu-PC or connected instruments.

Determines whether to enable or disable the external gain value.

Conditions Measurement views: All

Group Calibration commands

Syntax
```
CALibration:CORRection:EXTernal:GAIN:STATe { OFF | ON | 0 |
1 }
CALibration:CORRection:EXTernal:GAIN:STATe?
```

Related Commands CALibration:CORRection:EXTernal:GAIN[:MAGNitude]

Arguments OFF or 0 disables the external gain value.

ON or 1 enables the external gain value.

Examples CALIBRATION:CORRECTION:EXTERNAL:GAIN:STATEON enables the external gain value.

CALibration:CORRection:EXTernal:PROBe:CONNect? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Queries whether the external probe is connected to the analyzer or not.

Conditions Measurement views: All

Group Calibration commands

Syntax `CALibration:CORRection:EXTernal:PROBe:CONNect?`

Arguments None

Returns `{ 0 | 1 }`

`0` indicates that the external probe is not connected to the analyzer.

`1` indicates that the external probe is connected to the analyzer.

Examples `CALIBRATION:CORRECTION:EXTERNAL:PROBE:CONNECT?` might return `ON`, indicating that the external probe is connected to the analyzer.

CALibration:CORRection:EXTernal:PROBe[:MAGNitude]? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Queries the external probe attenuation.

Conditions Measurement views: All

Group Calibration commands

Syntax `CALibration:CORRection:EXTernal:PROBe[:MAGNitude]?`

Arguments None

Returns `<attenuation>::=<NRf>` The probe attenuation value in dB.

Examples `CALIBRATION:CORRECTION:EXTERNAL:PROBE:MAGNITUDE?` might return `10`, indicating that the probe attenuation is 10 dB.

CALibration:CORRection:EXTernal:PROBe:STATe

This command does not apply to SignalVu-PC or connected instruments.

Determines whether or not to correct data for the external probe attenuation.

Conditions Measurement views: All

Group Calibration commands

Syntax
```
CALibration:CORRection:EXTernal:PROBe:STATe { OFF | ON | 0 | 1 }
CALibration:CORRection:EXTernal:PROBe:STATe?
```

Arguments `OFF` or `0` does not correct data for the external probe attenuation.

`ON` or `1` corrects data for the external probe attenuation.

Examples `CALIBRATION:CORRECTION:EXTERNAL:PROBE:STATEON` corrects data for the external probe attenuation.

CALibration:CORRection:EXTernal:TYPE

This command does not apply to SignalVu-PC or connected instruments.

Sets or queries the data type to apply the external loss table corrections.

Conditions Measurement views: All

Group Calibration commands

Syntax
```
CALibration:CORRection:EXTernal:TYPE { TRACe | DATA }
CALibration:CORRection:EXTernal:TYPE
```

Related Commands CALibration:CORRection:EXTernal:GAIN:STATe

Arguments `TRACe` selects traces in the Spectrum, Spectrogram, Spurious, and Amplitude versus Time views. Selecting TRACe disables the CALibration:CORRection:EXTernal:GAIN:STATe command.

DATA selects all acquired data. Selecting DATA enables the CALibration:CORRection:EXTernal:GAIN:STATe command.

Examples CALIBRATION:CORRECTION:EXTERNAL:TYPETRACe selects traces to apply the external loss table corrections.

CALibration:TXGain:ABORt (No Query Form)

Cancels a calibration.

Conditions Measurement view: Transmission Gain, Perform Calibration window

Group Calibration group

Syntax CALibration:TXGain:ABORt

Arguments None.

Examples CALIBRATION:TXGAIN:ABORT cancels a calibration that is in process.

CALibration:TXGain[:ALL] (No Query Form)

Performs a calibration using the current settings.

Conditions Measurement view: Transmission Gain, Perform Calibration window

Group Calibration group

Syntax CALibration:TXGain[:ALL]

Arguments None.

Examples CALIBRATION:TXGAIN[:ALL] performs a calibration using the current settings.

CALibration:TXGain:FINish? (Query Only)

Queries whether or not the calibration is completed.

Conditions Measurement view: Transmission Gain, Perform Calibration window

Group Calibration group

Syntax `CALibration:TXGain:FINish?`

Returns 0 indicates calibration is in progress.

1 indicates calibration is completed.

Examples `CALIBRATION:TXGAIN:FINISH?` might return 1, indicating that the calibration is completed.

CALibration:TXGain:STATe? (Query Only)

Queries the calibration state for the Transmission Gain measurement.

Conditions Measurement view: Transmission Gain

Group Calibration group

Syntax `CALibration:TXGain:STATe?`

Returns CALIBRATED means that the unit has been calibrated and is making measurements at the same frequencies at which it was calibrated.

CALIBRATED? means that the unit has been calibrated, but one or more of the operating conditions has changed. Operating conditions are Tracking Generator Output Power, Reference Level, Preamp On/Off, RF Attenuation, or Temperature (>5 °C from the calibrated temperature).

INTERPOLATED means that the unit has been calibrated, but is making measurements at frequencies other than the calibration points and is interpolating the calibration between points.

Examples `CALIBRATION:TXGAIN:STATE?` might return CALIBRATED, indicating that the measurement has been calibrated with the current settings.

*CLS (No Query Form)

Clears the analyzer status data structures. Refer to Section 3, *Status and Events*, for the register information.

The *CLS command clears the following

- the Event Queue
- the Standard Event Status Register (SESR)
- the Status Byte Register (except the MAV bit; see below)

If the *CLS command immediately follows an <EOI>, the Output Queue and MAV bit (Status Byte Register bit 4) are also cleared. MAV indicates information is in the output queue. The device clear (DCL) GPIB control message will clear the output queue and thus MAV. *CLS does not clear the output queue or MAV. (A complete discussion of these registers and bits, and of event handling in general is described in the *Status and Events* section)

*CLS can suppress a Service Request that is to be generated by an *OPC. This will happen if a hardcopy output or single sequence acquisition operation is still being processed when the *CLS command is executed.

Conditions Measurement views: All

Group IEEE common commands

Syntax `*CLS`

Related Commands *ESE, *ESR?, *SRE, *STB?

Arguments None

Examples `*CLS` clears the analyzer status data structures.

DISPlay:ACPower:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the Channel power and ACPR view.

Conditions Measurement views: Channel power and ACPR

Group Display commands

Syntax
```
DISPlay:ACPower:MARKer:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:ACPower:MARKer:SHOW:STATe?
```

Arguments `OFF` or `0` hides the readout for the selected marker in the graph.

`ON` or `1` shows the readout for the selected marker in the graph.

Examples `DISPLAY:ACPOWER:MARKER:SHOW:STATEON` shows the readout for the selected marker in the graph.

DISPlay:ACPower:PLEVel:SHOW:STATe

Determines whether to show or hide the power levels in the Channel power and ACPR view.

Conditions Measurement views: Channel power and ACPR

Group Display commands

Syntax
```
DISPlay:ACPower:PLEVel:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:ACPower:PLEVel:SHOW:STATe?
```

Arguments `OFF` or `0` hides the power levels in the graph.

`ON` or `1` shows the power levels in the graph.

Examples `DISPLAY:ACPOWER:PLEVEL:SHOW:STATEON` shows the power levels in the graph.

DISPlay:ACPower:RESet:SCALe (No Query Form)

Resets the horizontal and vertical scale to the default values described below in the Channel power and ACPR view.

Vertical offset = Reference level,
Vertical scale = 100 dB,

Horizontal offset = Center frequency, and
Horizontal scale = Default span

Conditions Measurement views: Channel power and ACPR

Group Display commands

Syntax `DISPlay:ACPower:RESet:SCALe`

Arguments None

Examples `DISPLAY:ACPOWER:RESET:SCALE` resets the horizontal and vertical scale to the default values.

DISPlay:ACPower:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions Measurement views: Channel power and ACPR

Group Display commands

Syntax
```
DISPlay:ACPower:WINDow:TRACe:GRATicule:GRID:STATe { OFF |
ON | 0 | 1 }
DISPlay:ACPower:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples `DISPLAY:ACPOWER:WINDOW:TRACE:GRATICULE:GRID:STATEON` shows the graticule grid on the screen.

DISPlay:ACPower:X[:SCALe]

Sets or queries the horizontal range of the Channel power and ACPR graph.

Conditions Measurement views: Channel power and ACPR

Group Display commands

Syntax
```
DISPlay:ACPower:X[:SCALe] <value>
DISPlay:ACPower:X[:SCALe]?
```

Related Commands DISPlay:ACPower:X[:SCALe]:OFFSet

Arguments `<value>::=<NRf>` specifies the horizontal range.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `DISPLAY:ACPOWER:X:SCALE 10MHz` sets the horizontal range to 10 MHz.

DISPlay:ACPower:X[:SCALe]:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the Channel power and ACPR view.

Conditions Measurement views: Channel power and ACPR

Group Display commands

Syntax
```
DISPlay:ACPower:X[:SCALe]:AUTO
```

Arguments None

Examples `DISPLAY:ACPOWER:X:SCALE:AUTO` rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:ACPower:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (left edge) of the Channel power and ACPR graph.

Conditions Measurement views: Channel power and ACPR

Group Display commands

Syntax
```
DISPlay:ACPower:X[:SCALe]:OFFSet <value>
DISPlay:ACPower:X[:SCALe]:OFFSet?
```

Related Commands DISPlay:ACPower:X[:SCALe]

Arguments <value>::=<NRf> specifies the minimum horizontal value.
Range: [(center frequency) - (X scale) × 0.9] to [(center frequency) + (X scale) × 0.9]

Examples `DISPlay:ACPower:X:SCALe:OFFSet 1.45GHz` sets the minimum horizontal value to 1.45 GHz in the Channel power and ACPR graph.

DISPlay:ACPower:Y[:SCALe]

Sets or queries the vertical range of the channel power and ACPR graph.

Conditions Measurement views: Channel power and ACPR

Group Display commands

Syntax
```
DISPlay:ACPower:Y[:SCALe] <value>
DISPlay:ACPower:Y[:SCALe]?
```

Related Commands DISPlay:ACPower:Y[:SCALe]:OFFSet

Arguments <value>::=<NRf> specifies the vertical range. Range: 0.1 to 200 dB.

Examples `DISPLAY:ACPOWER:Y:SCALE 100` sets the vertical range to 100 dB in the Channel power and ACPR graph.

DISPlay:ACPower:Y[:SCALe]:AUTO (No Query Form)

Rescales the vertical axis automatically to fit the waveform to the screen in the Channel power and ACPR view.

Conditions Measurement views: Channel power and ACPR

Group Display commands

Syntax
```
DISPlay:ACPower:Y[:SCALe]:AUTO
```

Arguments None

Examples `DISPLAY:ACPOWER:Y:SCALE:AUTO` rescales the vertical scale automatically to fit the waveform to the screen.

DISPlay:ACPower:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the top edge of the vertical axis) in the Channel power and ACPR graph.

Conditions Measurement views: Channel power and ACPR

Group Display commands

Syntax
```
DISPlay:ACPower:Y[:SCALe]:OFFSet <value>
DISPlay:ACPower:Y[:SCALe]:OFFSet?
```

Related Commands DISPlay:ACPower:Y[:SCALe]

Arguments `<value>::=<NRf>` specifies the vertical offset. Range: -170 to +50 dBm.

Examples `DISPLAY:ACPOWER:Y:SCALE:OFFSET -12.5` sets the vertical offset to -12.5 dBm in the Channel power and ACPR graph.

DISPlay:ADEMod:MEASview:DELete (No Query Form)

Deletes the measurement view in the general purpose analog demodulation measurements.

Conditions Measurement views: General purpose analog demodulation

Group Display commands

Syntax `DISPlay:ADEMod:MEASview:DELete { AM | FM | PM }`

Arguments The following table lists the arguments.

Table 2-21: Analog demodulation measurement views

Argument	View
AM	AM measurement (modulation factor versus time)
FM	FM measurement (frequency deviation versus time)
PM	PM measurement (phase deviation versus time)

If you attempt to delete a view that is not displayed on screen, the error (-200, "Execution error; Measurement is not running") will be returned.

Examples `DISPLAY:ADEMOD:MEASVIEW:DELETEAM` deletes the AM measurement view.

DISPlay:ADEMod:MEASview:NEW (No Query Form)

Displays a new measurement view in the general purpose analog demodulation measurements.

Conditions Measurement views: General purpose analog demodulation

Group Display commands

Syntax `DISPlay:ADEMod:MEASview:NEW { AM | FM | PM }`

Arguments (See Table 2-21 on page 2-408.) If you attempt to open a view that is currently displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Examples `DISPLAY:ADEMOD:MEASVIEW:NEWAM` creates the AM measurement view.

DISPlay:ADEMod:MEASview:SELect

Selects a measurement view on the screen in the general purpose analog demodulation measurements. The query command returns the currently selected view.

Selecting a measurement optimizes it. Other measurements may be optimized as a side effect. Refer to the DISPlay:WINDow:OPTimized:MEASurement? query.

Conditions Measurement views: General purpose analog demodulation

Group Display commands

Syntax

```
DISPlay:ADEMod:MEASview:SELect { AM | FM | PM }
DISPlay:ADEMod:MEASview:SELect?
```

Arguments (See Table 2-21 on page 2-408.) If you attempt to select a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running") will be returned.

Examples `DISPLAY:ADEMOD:MEASVIEW:SELECTAM` selects the AM measurement view.

DISPlay:{AM|FM|PM}:MARKer:SHOW:STATe

Determines whether to show or hide the marker readout in the AM/FM/PM measurement view.

Conditions Measurement views: General purpose analog demodulation

Group Display commands

Syntax

```
DISPlay:{AM|FM|PM}:MARKer:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:{AM|FM|PM}:MARKer:SHOW:STATe?
```

Arguments `OFF` or `0` hides the marker readout on the screen.

`ON` or `1` shows the marker readout on the screen.

Examples `DISPLAY:AM:MARKER:SHOW:STATE ON` shows the marker readout in the AM measurement view.

DISPlay:{AM|FM|PM}:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions Measurement views: General purpose analog demodulation

Group Display commands

Syntax

```
DISPlay:{AM|FM|PM}:WINDow:TRACe:GRATicule:GRID:STATe { OFF |
ON | 0 | 1 }
DISPlay:{AM|FM|PM}:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples `DISPLAY:AM:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the AM measurement view.

DISPlay:{AM|FM|PM}:X:RSCale (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the AM/FM/PM measurement display.

Conditions Measurement views: General purpose analog demodulation

Group Display commands

Syntax `DISPlay:{AM|FM|PM}:X:RSCale`

Arguments None

Examples `DISPLAY:AM:X:RSCALE` rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:{AM|FM|PM}:X[:SCALe]:AUTO (No Query Form)

Sets the horizontal scale automatically to fit the waveform to the screen in the AM/FM/PM measurement display.

Conditions Measurement views: General purpose analog demodulation

Group Display commands

Syntax `DISPlay:{AM|FM|PM}:X[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:AM:X:SCALE:AUTO` sets the horizontal scale automatically to fit the waveform to the screen in the AM/FM/PM measurement display.

DISPlay:{AM|FM|PM}:X[:SCALe]:FULL

Sets or queries the horizontal scale (full-scale time) of the AM/FM/PM measurement graph.

Conditions Measurement views: General purpose analog demodulation

Group Display commands

Syntax
```
DISPlay:{AM|FM|PM}:X[:SCALe]:FULL <value>
DISPlay:{AM|FM|PM}:X[:SCALe]:FULL?
```

Arguments `<value>::=<NRf>` specifies the horizontal scale in full-scale time.
Range: 0 to 10^{27} s.

Examples `DISPlay:AM:X:SCALe:FULL 35us` sets the horizontal scale to 35 µs in the AM measurement.

DISPlay:{AM|FM|PM}:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (left edge) of the AM/FM/PM measurement graph.

Conditions Measurement views: General purpose analog demodulation

Group Display commands

Syntax
```
DISPlay:{AM|FM|PM}:X[:SCALe]:OFFSet <value>
DISPlay:{AM|FM|PM}:X[:SCALe]:OFFSet?
```

Arguments <value>::=<NRf> specifies the minimum horizontal value.
Range: –0.9×(horizontal scale) to +0.9×(horizontal scale)

Examples `DISPLAY:AM:X:SCALE:OFFSET 20.075us` sets the minimum horizontal value to 20.075 µs.

DISPlay:{AM|FM|PM}:Y:RSCale (No Query Form)

Rescales the vertical axis automatically to fit the waveform to the screen in the AM/FM/PM measurement display.

Conditions Measurement views: General purpose analog demodulation

Group Display commands

Syntax
```
DISPlay:{AM|FM|PM}:Y:RSCale
```

Arguments None

Examples `DISPLAY:AM:Y:RSCALE` rescales the vertical scale automatically to fit the waveform to the screen in the AM measurement.

DISPlay:{AM|FM|PM}:Y[:SCALe]

Sets or queries the vertical range of the AM/FM/PM measurement graph.

Conditions Measurement views: General purpose analog demodulation

Group Display commands

Syntax

```
DISPlay:{AM|FM|PM}:Y[:SCALe] <value>
DISPlay:{AM|FM|PM}:Y[:SCALe]?
```

Arguments

<value>::=<NRf> specifies the vertical range.
The setting range depends on measurements as shown in the following table.

Setting range

Measurement	Setting range
AM	1% to 500%
FM	10 Hz to 120 MHz
PM	10° to 400°

Examples

`DISPLAY:AM:Y:SCALE 75` sets the vertical range to 75% in the AM measurement graph.

DISPlay:{AM|FM|PM}:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (center point of the vertical axis) in the AM/FM/PM measurement graph.

Conditions

Measurement views: General purpose analog demodulation

Group

Display commands

Syntax

```
DISPlay:{AM|FM|PM}:Y[:SCALe]:OFFSet <value>
DISPlay:{AM|FM|PM}:Y[:SCALe]:OFFSet?
```

Arguments

<value>::=<NRf> specifies the vertical offset.
The setting range depends on measurements as shown in the following table.

Setting range

Measurement	Setting range
AM	–500% to +500%
FM	–(maxVertScale * (1 + 1.1) * 0.5) to +(maxVertScale * (1 + 1.1) * 0.5)
PM	–180° to +180°

Examples

`DISPLAY:AM:Y:SCALE:OFFSET -12.5` sets the vertical offset to –12.5% in the AM measurement graph.

DISPlay:ANTenna:AZIMuth:SHOW:STATe

The command checks the Show Compass box in the UI to display the compass. The query form returns whether or not the Show Compass box is checked.

Group Display commands

Syntax
```
DISPlay:ANTenna:AZIMuth:SHOW:STATe { 1 | 0 | ON | OFF }
DISPlay:ANTenna:AZIMuth:SHOW:STATe?
```

Arguments 1 or ON sets compass to show.

0 or OFF sets compass to hide.

Returns 1 or ON means compass is set to show.

0 or OFF means compass is set to hide.

DISPlay:AUDio:MEASview:DELete (No Query Form)

Deletes the specified audio analysis view.

Conditions Measurement views: Audio Spectrum, Audio Summary

Group Display commands

Syntax
```
DISPlay:AUDio:MEASview:DELete { SPECtrum | SUMMary }
```

Arguments `SPECtrum` deletes the Audio Spectrum view.

`SUMMary` deletes the Audio Summary view.

If you attempt to delete a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running") will be returned.

Examples `DISPLAY:AUDIO:MEASVIEW:DELETE SUMMary` deletes the Audio Summary view.

DISPlay:AUDio:MEASview:NEW (No Query Form)

Displays a new audio analysis view.

Conditions Measurement views: Audio Spectrum, Audio Summary

Group Display commands

Syntax `DISPlay:AUDio:MEASview:NEW { SPECtrum | SUMMary }`

Arguments `SPECtrum` creates a new Audio Spectrum view.

`SUMMary` creates a new Audio Summary view.

If you attempt to open a view that is currently displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Examples `DISPLAY:AUDIO:MEASVIEW:NEW SUMMary` creates a new Audio Summary view.

DISPlay:AUDio:MEASview:SELect

Selects an audio analysis view on the screen. The query returns the currently selected view.

Selecting a measurement optimizes it. Other measurements may be optimized as a side effect. Refer to the DISPlay:WINDow:OPTimized:MEASurement? query.

Conditions Measurement views: Audio Spectrum, Audio Summary

Group Display commands

Syntax
```
DISPlay:AUDio:MEASview:SELect { SPECtrum | SUMMary }
DISPlay:AUDio:MEASview:SELect?
```

Arguments `SPECtrum` creates a new Audio Spectrum view.

`SUMMary` creates a new Audio Summary view.

If you attempt to open a view that is currently displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Examples `DISPLAY:AUDIO:MEASVIEW:SELECT SPECTrum` selects the Audio Spectrum display.

DISPlay:AUDio:SPECtrum:FREQuency:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the audio spectrum view.

Conditions Measurement views: Audio Spectrum

Group Display commands

Syntax `DISPlay:AUDio:SPECtrum:FREQuency:AUTO`

Arguments None

Examples `DISPLAY:AUDIO:SPECTRUM:FREQUENCY:AUTO` rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:AUDio:SPECtrum:FREQuency:[SCALe]:STARt

Sets or queries the start frequency (left edge) of the audio spectrum graph.

Conditions Measurement views: Audio Spectrum

Group Display commands

Syntax
```
DISPlay:AUDio:SPECtrum:FREQuency:[SCALe]:STARt <value>
DISPlay:AUDio:SPECtrum:FREQuency:[SCALe]:STARt?
```

Related Commands DISPlay:AUDio:SPECtrum:FREQuency:[SCALe]:STOP

Arguments <value> ::= <NRf> specifies the start frequency.

Examples `DISPLAY:AUDIO:SPECTRUM:FREQUENCY:[SCALE]:START` 0 sets the start frequency to 0 Hz.

DISPlay:AUDio:SPECtrum:FREQuency:[SCALe]:STOP

Sets or queries the stop frequency (right edge) of the audio spectrum graph.

Conditions Measurement views: Audio Spectrum

Group Display commands

Syntax
```
DISPlay:AUDio:SPECtrum:FREQuency:[SCALe]:STOP <value>
DISPlay:AUDio:SPECtrum:FREQuency:[SCALe]:STOP?
```

Related Commands DISPlay:AUDio:SPECtrum:FREQuency:[SCALe]:STARt

Arguments <value> ::= <NRf> specifies the stop frequency.

Examples `DISPLAY:AUDIO:SPECTRUM:FREQUENCY:[SCALE]:STOP 20E+3` sets the stop frequency to 20 kHz.

DISPlay:AUDio:SPECtrum:MARKer:SHOW:STATe

Determines whether to show or hide the readouts for the selected marker in the audio spectrum view.

Conditions Measurement views: Audio Spectrum

Group Display commands

Syntax
```
DISPlay:AUDio:SPECtrum:MARKer:SHOW:STATe { OFF | ON | 0 |
1 }
DISPlay:AUDio:SPECtrum:MARKer:SHOW:STATe
```

Arguments `OFF` or `0` hides the readout for the selected marker in the view.

`ON` or `1` shows the readout for the selected marker in the view.

Examples `DISPLAY:AUDIO:SPECTRUM:MARKER:SHOW:STATE ON` shows the readout for the selected marker in the audio spectrum view.

DISPlay:AUDio:SPECtrum:RESet:SCALe (No Query Form)

Resets the horizontal and vertical scales in the audio spectrum view.

Conditions Measurement views: Audio Spectrum

Group Display commands

Syntax `DISPlay:AUDio:SPECtrum:RESet:SCALe`

Arguments None

Examples `DISPLAY:AUDIO:SPECTRUM:RESET:SCALE` resets the horizontal and vertical scales.

DISPlay:AUDio:SPECtrum:SCALe:LOG:STATe

Determines whether or not to set the horizontal axis logarithmic in the audio spectrum view.

Conditions Measurement views: Audio Spectrum

Group Display commands

Syntax
```
DISPlay:AUDio:SPECtrum:SCALe:LOG:STATe { OFF | ON | 0 | 1 }
DISPlay:AUDio:SPECtrum:SCALe:LOG:STATe?
```

Arguments `OFF` or `0` sets the horizontal axis linear.

`ON` or `1` sets the horizontal axis logarithmic.

Examples `DISPLAY:AUDIO:SPECTRUM:SCALE:LOG:STATE` ON sets the horizontal axis logarithmic in the audio spectrum view.

DISPlay:AUDio:SPECtrum:SHOW:NHARmonic:THReshold

Sets or queries the appearance of the non-harmonic threshold in the audio spectrum view.

Conditions Measurement views: Audio Spectrum

Group Display commands

Syntax
```
DISPlay:AUDio:SPECtrum:SHOW:NHARmonic:THReshold { SHADed | LINE | OFF }
DISPlay:AUDio:SPECtrum:SHOW:NHARmonic:THReshold?
```

Arguments `SHADed` uses shading for the non-harmonic threshold area in the display.

`LINE` uses a line for the non-harmonic threshold area.

`OFF` turns off the non-harmonic threshold in the display.

Examples `DISPLAY:AUDIO:SPECTRUM:SHOW:NHARMONIC:THRESHOLD SHADed` uses shading to identify the non-harmonic threshold area in the audio spectrum view.

DISPlay:AUDio:SPECtrum:TABLe:SHOW:STATe

Determines whether to show or hide the results table in the audio spectrum view.

Conditions Measurement views: Audio Spectrum

Group Display commands

Syntax
```
DISPlay:AUDio:SPECtrum:TABLe:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:AUDio:SPECtrum:TABLe:SHOW:STATe?
```

Arguments `OFF` or `0` disables (hides) the results table in the audio spectrum view.

`ON` or `1` shows the results table in the audio spectrum view.

Examples `DISPLAY:AUDIO:SPECTRUM:TABLE:SHOW:STATE OFF` hides the results table in the audio spectrum view.

DISPlay:AUDio:SPECtrum:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions Measurement views: Audio Spectrum

Group Display commands

Syntax

```
DISPlay:AUDio:SPECtrum:WINDow:TRACe:GRATicule:GRID:STATe {
OFF | ON | 0 | 1 }
DISPlay:AUDio:SPECtrum:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

OFF or 0 hides the graticule grid.

ON or 1 shows the graticule grid.

Examples

DISPLAY:AUDIO:SPECTRUM:WINDOW:TRACE:GRATICULE:GRID:STATE OFF hides the graticule grid on the screen.

DISPlay:AUDio:SPECtrum:Y:[SCALe]

Sets or queries the vertical scale in the audio spectrum view.

The range of the scale is from 0.1 dB to 200 dB.

Conditions

Measurement views: Audio Spectrum

Group

Display commands

Syntax

```
DISPlay:AUDio:SPECtrum:Y:[SCALe] <value>
DISPlay:AUDio:SPECtrum:Y:[SCALe]?
```

Arguments

<value> ::= <NRf> specifies the vertical scale value in dB.

Examples

DISPLAY:AUDIO:SPECTRUM:Y:[SCALE] 100 sets the vertical scale to 100 dB.

DISPlay:AUDio:SPECtrum:Y:[SCALe]:AUTO (No Query Form)

Rescales the vertical axis automatically in the audio spectrum view.

Conditions

Measurement views: Audio Spectrum

Group

Display commands

Syntax

```
DISPlay:AUDio:SPECtrum:Y:[SCALe]:AUTO
```

Arguments None

Examples `DISPLAY:AUDIO:SPECTRUM:Y:[SCALE]:AUTO` rescales the vertical axis automatically.

DISPlay:AUDio:SPECtrum:Y:[SCALe]:OFFSet

Sets or queries the vertical position (offset) in the audio spectrum view.

Conditions Measurement views: Audio Spectrum

Group Display commands

Syntax
```
DISPlay:AUDio:SPECtrum:Y:[SCALe]:OFFSet <value>
DISPlay:AUDio:SPECtrum:Y:[SCALe]:OFFSet?
```

Arguments <value> ::= <NRf> specifies the vertical offset in dBm.

Examples `DISPLAY:AUDIO:SPECTRUM:Y:[SCALE]:OFFSET` 150 sets the vertical offset to 150 dBM².

DISPlay:AVTime:LEGend:STATe

Determines whether to show or hide the trace legend in the amplitude versus time view. The legend indicates the trace detection and function on the screen for each displayed trace.

Conditions Measurement views: Amplitude versus Time

Group Display commands

Syntax
```
DISPlay:AVTime:LEGend:STATe { OFF | ON | 0 | 1 }
DISPlay:AVTime:LEGend:STATe?
```

Arguments `OFF` or `0` hides the trace legend.

`ON` or `1` shows the trace legend.

Examples `DISPLAY:AVTIME:LEGEND:STATEON` shows the trace legend on the screen.

DISPlay:AVTime:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker on the screen in the Amplitude versus Time measurement.

Conditions Measurement views: Amplitude versus Time

Group Display commands

Syntax
```
DISPlay:AVTime:MARKer:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:AVTime:MARKer:SHOW:STATe?
```

Arguments `OFF` or `0` hides the readout for the selected marker on screen.

`ON` or `1` shows the readout for the selected marker on screen.

Examples `DISPLAY:AVTIME:MARKER:SHOW:STATEON` shows the readout for the selected marker on screen.

DISPlay:AVTime:RESet (No Query Form)

Resets the horizontal and vertical scale to the default values described below in the Amplitude versus Time view.

Vertical offset = Reference level,
Vertical scale = 100 dB,
Horizontal offset = Analysis offset, and
Horizontal scale = Analysis length

Conditions Measurement views: Amplitude versus Time

Group Display commands

Syntax
```
DISPlay:AVTime:RESet
```

Arguments None

Examples `DISPLAY:AVTIME:RESET` resets the horizontal and vertical scale to the default values.

DISPlay:AVTime:TRIGger:LEVel:STATe

Determines whether to show or hide the power trigger level line on the screen in the Amplitude versus Time measurement.

Conditions Measurement views: Amplitude versus Time

Group Display commands

Syntax
```
DISPlay:AVTime:TRIGger:LEVel:STATe { OFF | ON | 0 | 1 }
DISPlay:AVTime:TRIGger:LEVel:STATe?
```

Arguments `OFF` or `0` hides the power trigger level line.

`ON` or `1` shows the power trigger level line.

Examples `DISPLAY:AVTIME:TRIGGER:LEVEL:STATEON` shows the power trigger level line on the screen.

DISPlay:AVTime:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions Measurement views: Amplitude versus Time

Group Display commands

Syntax
```
DISPlay:AVTime:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON
| 0 | 1 }
DISPlay:AVTime:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples `DISPLAY:AVTIME:WINDOW:TRACE:GRATICULE:GRID:STATEON` shows the graticule grid on the screen in the Amplitude versus Time view.

DISPlay:AVTime:X:RSCale (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the Amplitude versus Time display.

Conditions Measurement views: Amplitude versus Time

Group Display commands

Syntax `DISPlay:AVTime:X:RSCale`

Arguments None

Examples `DISPLAY:AVTIME:X:RSCALE` rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:AVTime:X[:SCALe]:AUTO (No Query Form)

Sets the horizontal scale automatically to fit the waveform to the screen in the Amplitude versus Time view. Executing this command sets DISPlay:AVTime:X[:SCALe]:AUTO:STATe ON.

Conditions Measurement views: Amplitude versus Time

Group Display commands

Syntax `DISPlay:AVTime:X[:SCALe]:AUTO`

Related Commands DISPlay:AVTime:X[:SCALe]:AUTO:STATe

Arguments None

Examples `DISPlay:AVTime:X:SCALe:AUTO` sets the horizontal scale automatically to fit the waveform to the screen.

DISPlay:AVTime:X[:SCALe]:AUTO:STATe

Determines whether to set the horizontal scale automatically or manually.

Conditions Measurement views: Amplitude versus Time

Group Display commands

Syntax

```
DISPlay:AVTime:X[:SCALe]:AUTO:STATe { OFF | ON | 0 | 1 }
DISPlay:AVTime:X[:SCALe]:AUTO:STATe?
```

Arguments `OFF` or `0` specifies that the horizontal scale is set manually. To set it, use the DISPlay:AVTime:X[:SCALe]:FULL and DISPlay:AVTime:X[:SCALe]:OFFSet commands.

`ON` or `1` specifies that the horizontal scale is set automatically.

Examples `DISPLAY:AVTIME:X:SCALE:AUTO:STATE ON` specifies that the horizontal scale is set automatically.

DISPlay:AVTime:X[:SCALe]:FULL

Sets or queries the horizontal scale (full-scale time) of the Amplitude versus Time graph. Programming a specified scale sets DISPlay:AVTime:X[:SCALe] AUTO:STATe OFF.

Conditions Measurement views: Amplitude versus Time

Group Display commands

Syntax

```
DISPlay:AVTime:X[:SCALe]:FULL <value>
DISPlay:AVTime:X[:SCALe]:FULL?
```

Related Commands DISPlay:AVTime:X[:SCALe]:AUTO:STATe, DISPlay:AVTime:X[:SCALe]:OFFSet

Arguments `<value>::=<NRf>` specifies the horizontal scale in full-scale time. Use the DISPlay:AVTime:X[:SCALe]:MAXimum? and DISPlay:AVTime:X[:SCALe]:MINimum? queries to get the upper and lower limits of the setting range.

Examples `DISPLAY:AVTIME:X:SCALE:FULL 25.6us` sets the horizontal scale to 25.6 µs.

DISPlay:AVTime:X[:SCALe]:MAXimum? (Query Only)

Queries the upper limit of the horizontal scale setting range.

Conditions Measurement views: Amplitude versus Time

Group Display commands

Syntax `DISPlay:AVTime:X[:SCALe]:MAXimum?`

Related Commands DISPlay:AVTime:X[:SCALe]:FULL

Arguments None

Returns `<NRf>` The upper limit of the horizontal scale setting range.

Examples `DISPLAY:AVTIME:X:SCALE:MAXIMUM?` might return `18.135E-3`, indicating that the upper limit of the horizontal scale setting range is 18.135 ms.

DISPlay:AVTime:X[:SCALe]:MINimum? (Query Only)

Queries the lower limit of the horizontal scale setting range.

Conditions Measurement views: Amplitude versus Time

Group Display commands

Syntax `DISPlay:AVTime:X[:SCALe]:MINimum?`

Related Commands DISPlay:AVTime:X[:SCALe]:FULL

Arguments None

Returns <NRf> The lower limit of the horizontal scale setting range.

Examples DISPLAY:AVTIME:X:SCALE:MINIMUM? might return 10.0E-9, indicating that the lower limit of the horizontal scale setting range is 10.0 ns.

DISPlay:AVTime:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (left edge) of the Amplitude versus Time graph. Programming a specified offset sets DISPlay:AVTime:X[:SCALe] AUTO:STATe OFF.

Conditions Measurement views: Amplitude versus Time

Group Display commands

Syntax
```
DISPlay:AVTime:X[:SCALe]:OFFSet <value>
DISPlay:AVTime:X[:SCALe]:OFFSet?
```

Related Commands DISPlay:AVTime:X[:SCALe]:AUTO:STATe, DISPlay:AVTime:X[:SCALe]:FULL

Arguments <value>::=<NRf> specifies the minimum horizontal value. Use the DISPlay:AVTime:X[:SCALe]:OFFSet:MAXimum? and DISPlay:AVTime:X[:SCALe]:OFFSet:MINimum? queries to get the upper and lower limits of the setting range.

Examples DISPLAY:AVTIME:X:SCALE:OFFSET 800ns sets the minimum horizontal value to 800 ns in the Amplitude versus Time graph.

DISPlay:AVTime:X[:SCALe]:OFFSet:MAXimum? (Query Only)

Queries the upper limit of the horizontal offset setting range.

Conditions Measurement views: Amplitude versus Time

Group Display commands

Syntax
```
DISPlay:AVTime:X[:SCALe]:OFFSet:MAXimum?
```

Related Commands DISPlay:AVTime:X[:SCALe]:OFFSet

Arguments None

Returns <NRf> The upper limit of the horizontal offset setting range.

Examples DISPLAY:AVTIME:X:SCALE:OFFSET:MAXIMUM? might return -1.812E-3, indicating that the upper limit of the horizontal offset setting range is -1.812 ms.

DISPlay:AVTime:X[:SCALe]:OFFSet:MINimum? (Query Only)

Queries the lower limit of the horizontal offset setting range.

Conditions Measurement views: Amplitude versus Time

Group Display commands

Syntax DISPlay:AVTime:X[:SCALe]:OFFSet:MINimum?

Related Commands DISPlay:AVTime:X[:SCALe]:OFFSet

Arguments None

Returns <NRf> The lower limit of the horizontal offset setting range.

Examples DISPLAY:AVTIME:X:SCALE:OFFSET:MINIMUM? might return -16.28E-3, indicating that the lower limit of the horizontal offset setting range is -16.28 ms.

DISPlay:AVTime:Y:RSCale (No Query Form)

Rescales the vertical axis automatically to fit the waveform to the screen in the Amplitude versus Time display.

Conditions Measurement views: Amplitude versus Time

Group Display commands

Syntax `DISPlay:AVTime:Y:RSCale`

Arguments None

Examples `DISPLAY:AVTIME:Y:RSCALE` rescales the vertical scale automatically to fit the waveform to the screen.

DISPlay:AVTime:Y[:SCALe]:FULL

Sets or queries the vertical range of the Amplitude versus Time graph.

Conditions Measurement views: Amplitude versus Time

Group Display commands

Syntax
```
DISPlay:AVTime:Y[:SCALe]:FULL <value>
DISPlay:AVTime:Y[:SCALe]:FULL?
```

Related Commands DISPlay:AVTime:Y[:SCALe]:OFFSet

Arguments `<value>::=<NRf>` specifies the vertical range. Range: 0.1 to 200 dB.

Examples `DISPLAY:AVTIME:Y:SCALE:FULL 100` sets the vertical range to 100 dB in the Amplitude versus Time graph.

DISPlay:AVTime:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the top edge of the vertical axis) in the Amplitude versus Time graph.

Conditions Measurement views: Amplitude versus Time

Group Display commands

Syntax
```
DISPlay:AVTime:Y[:SCALe]:OFFSet <value>
DISPlay:AVTime:Y[:SCALe]:OFFSet?
```

Related Commands DISPlay:AVTime:Y[:SCALe]:FULL

Arguments <value>::=<NRf> specifies the vertical offset. Range: -170 to +50 dBm.

Examples DISPLAY:AVTIME:Y:SCALE:OFFSET -12.5 sets the vertical offset to -12.5 dBm in the Amplitude versus Time graph.

DISPlay:BIBEmissions:MARKer<x>:SHOW:STATe

Sets or queries to show or hide the readout for the selected marker Bluetooth InBand Emission view.

The parameter <x> = 1 to 4; Marker 0 (reference marker) is invalid. The specified marker must be activated using the CALCulate:MARKer:ADD command.

Conditions Measurement views: Bluetooth InBand Emission

Group Display commands

Syntax
```
DISPlay:BIBEmissions:MARKer<x>:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:BIBEmissions:MARKer<x>:SHOW:STATe?
```

Arguments OFF or 0 hides the readout for the selected marker in the graph.

ON or 1 shows the readout for the selected marker in the graph.

Examples DISPLAY:BIBEMISSIONS:MARKER1:SHOW:STATE ON shows the readout for the Marker1 (M1) in the graph.

DISPlay:BIBEmissions:PLEVel:SHOW:STATe

Determines whether to show or hide the power levels in the Bluetooth InBand Emission view.

Conditions Measurement views: Bluetooth InBand Emission

Group Display commands

Syntax

```
DISPlay:BIBEmissions:PLEVel:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:BIBEmissions:PLEVel:SHOW:STATe?
```

Arguments

OFF or 0 hides the power levels in the graph.

ON or 1 shows the power levels in the graph.

Examples

`DISPLAY:BIBEMISSIONS:PLEVEL:SHOW:STATE`ON shows the power levels in the graph.

DISPlay:BIBEmissions:RESet:SCALe (No Query Form)

Resets the horizontal and vertical scale to the default values in the Bluetooth InBand Emission view.

Conditions

Measurement views: Bluetooth InBand Emission

Group

Display commands

Syntax

```
DISPlay:BIBEmissions:RESet:SCALe
```

Arguments

None

Examples

`DISPLAY:BIBEMISSIONS:RESET:SCALE` resets the horizontal and vertical scale to the default values.

DISPlay:BIBEmissions:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries the graticule hidden or showing state.

Conditions

Measurement views: Bluetooth InBand Emission

Group

Display commands

Syntax

```
DISPlay:BIBEmissions:WINDow:TRACe:GRATicule:GRID:STATe { OFF
| ON | 0 | 1 }
DISPlay:BIBEmissions:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments OFF or 0 hides the graticule grid.

ON or 1 shows the graticule grid.

Examples `DISPLAY:BIBEMISSIONS:WINDOW:TRACE:GRATICULE:GRID:STATEON` shows the graticule grid on the screen.

DISPlay:BIBEmissions:X[:SCALe]

Sets or queries the horizontal range of the Bluetooth InBand Emission graph.

Conditions Measurement views: Bluetooth InBand Emission

Group Display commands

Syntax
```
DISPlay:BIBEmissions:X[:SCALe] <value>
DISPlay:BIBEmissions:X[:SCALe]?
```

Related Commands DISPlay:BIBEmissions:Y[:SCALe]:OFFSet

Arguments <value>::=<NRf> specifies the horizontal range.

Examples `DISPLAY:BIBEMISSIONS:X:SCALE 10MHZ` sets the horizontal range to 10 MHz.

DISPlay:BIBEmissions:X[:SCALe]:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the Bluetooth InBand Emission view.

Conditions Measurement views: Bluetooth InBand Emission

Group Display commands

Syntax `DISPlay:BIBEmissions:X[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:BIBEMISSIONS:X:SCALE:AUTO` rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:BIBEmissions:X[:SCALe]:OFFSet

Sets or queries the center frequency of the Bluetooth InBand Emission graph.

Conditions Measurement views: Bluetooth InBand Emission

Group Display commands

Syntax
```
DISPlay:BIBEmissions:X[:SCALe]:OFFSet <value>
DISPlay:BIBEmissions:X[:SCALe]:OFFSet?
```

Related Commands DISPlay:BIBEmissions:X[:SCALe]

Arguments <value> ::= <NRf> specifies the minimum horizontal value.

Examples `DISPLAY:BIBEMISSIONS:X:SCALE:OFFSET 1.45GHZ` sets the center frequency to 1.45 GHz.

DISPlay:BIBEmissions:Y[:SCALe]

Sets or queries the vertical range of the Bluetooth InBand Emission graph.

Conditions Measurement views: Bluetooth InBand Emission

Group Display commands

Syntax
```
DISPlay:BIBEmissions:Y[:SCALe] <value>
DISPlay:BIBEmissions:Y[:SCALe]?
```

Related Commands DISPlay:BIBEmissions:Y[:SCALe]:OFFSet

Arguments <value> ::= <NRf> specifies the vertical range. Range: 0.1 to 200 dB.

Examples `DISPLAY:BIBEMISSIONS:Y:SCALE 100` sets the vertical range to 100 dB.

DISPlay:BIBEmissions:Y[:SCALe]:AUTO (No Query Form)

Rescales the vertical axis automatically to fit the waveform to the screen in the Bluetooth InBand Emission view.

Conditions Measurement views: Bluetooth InBand Emission

Group Display commands

Syntax `DISPlay:BIBEmissions:Y[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:BIBEMISSIONS:Y:SCALE:AUTO` rescales the vertical scale automatically to fit the waveform to the screen.

DISPlay:BIBEmissions:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the top edge of the vertical axis) in the Bluetooth InBand Emission graph.

Conditions Measurement views: Bluetooth InBand Emission

Group Display commands

Syntax
```
DISPlay:BIBEmissions:Y[:SCALe]:OFFSet <value>
DISPlay:BIBEmissions:Y[:SCALe]:OFFSet?
```

Related Commands DISPlay:BIBEmissions:Y[:SCALe]

Arguments <value> ::= <NRf> specifies the vertical offset. Range: -170 to +50 dBm.

Examples DISPLAY:BIBEMISSIONS:Y:SCALE:OFFSET -12.5 sets the vertical offset to -12.5 dBm.

DISPlay:BLUEtooth:CONSte:TRACe:GRATicule:GRID:STATe

Sets or queries the display state of Constellation graticule grid.

Conditions Measurement views: Bluetooth Constellation

Group Display commands

Syntax
```
DISPlay:BLUEtooth:CONSte:TRACe:GRATicule:GRID:STATe
{0|1|OFF|ON}
DISPlay:BLUEtooth:CONSte:TRACe:GRATicule:GRID:STATe?
```

Arguments ON or 1 turns on the grid.
OFF or 0 turns off the grid.

Examples DISPlay:BLUEtooth:CONSte:TRACe:GRATicule:GRID:STATe 1 turns on the grid in the Constellation plot.

DISPlay:BLUEtooth:CONSte:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries whether to show or hide the graticule grid on the screen in the Bluetooth Constellation display.

Conditions Measurement views: Bluetooth Constellation

Group Display commands

Syntax
```
DISPlay:BLUEtooth:CONSte:WINDow:TRACe:GRATicule:GRID:STATe
STATe { OFF | ON | 0 | 1 }
DISPlay:BLUEtooth:CONSte:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments OFF or 0 hides the graticule grid.
ON or 1 shows the graticule grid.

Examples `DISPlay:BLUEtooth:CONSte:WINDow:TRACe:GRATicule:GRID:STATe ON` shows the graticule grid on the screen.

DISPlay:BLUEtooth:EDIagram:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries graticule grid hidden or showing state in the Bluetooth eye diagram display.

Conditions Measurement views: Bluetooth Eye diagram

Group Display commands

Syntax
```
DISPlay:BLUEtooth:EDIagram:WINDow:TRACe:GRATicule:GRID:STATe
{0|1|OFF|ON}
DISPlay:BLUEtooth:EDIagram:WINDow:TRACe:GRATicule:GRID:
STATe?
```

Arguments OFF or 0 hides the graticule grid.
ON or 1 shows the graticule grid.

Examples
```
DISPLAY:BLUETOOTH:EDIAGRAM:WINDOW:TRACE:GRATICULE:GRID:STATE
```
ON shows the graticule grid in the eye diagram.

DISPlay:BLUEtooth:EDIagram:Y[:SCALe]

Sets or queries the vertical range of the Bluetooth eye diagram.

Conditions Measurement views: Bluetooth Eye diagram

Group Display commands

Syntax
```
DISPlay:BLUEtooth:EDIagram:Y[:SCALe] <value>
DISPlay:BLUEtooth:EDIagram:Y[:SCALe]?
```

Arguments <value> ::= <NRf> the vertical range (1 to 100)

Examples `DISPlay:BLUEtooth:EDIagram:Y:SCALe 2.5` sets the vertical range to 2.5 in the eye diagram.

DISPlay:BLUEtooth:EDIagram:Y[:SCALe]:AUTO (No Query Form)

Sets the vertical scale automatically to fit the waveform in the Bluetooth eye diagram.

Conditions Measurement views: Bluetooth Eye diagram

Group Display commands

Syntax `DISPlay:BLUEtooth:EDIagram:Y[:SCALe]:AUTO`

Arguments None

Returns `DISPLAY:BLUETOOTH:EDIAGRAM:Y:SCALE:AUTO` sets the vertical scale automatically to fit the waveform to the screen.

DISPlay:BLUEtooth:EDIagram:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (center point of the vertical axis) of the Bluetooth eye diagram.

Conditions Measurement views: Bluetooth Eye diagram

Group Display commands

Syntax
```
DISPlay:BLUEtooth:EDIagram:Y[:SCALe]:OFFSet <value>
DISPlay:BLUEtooth:EDIagram:Y[:SCALe]:OFFSet?
```

Arguments <value> ::= <NRf> the minimum vertical scale offset, range (-50 to +50).

Examples `DISPlay:BLUEtooth:EDIagram:Y:SCALe:OFFSet -0.5` sets the vertical offset to -0.5.

DISPlay:BLUEtooth:FDVTime:WINDow:TRACe:GRATicule:GRID:STATe

Shows or hides the graticule grid in the Bluetooth Frequency Deviation vs. Time display.

Conditions Measurement views: Frequency Deviation vs. Time

Group Display commands

Syntax
```
DISPlay:BLUEtooth:FDVTime:WINDow:TRACe:GRATicule:GRID:STATe
{OFF | ON | 0 | 1}
DISPlay:BLUEtooth:FDVTime:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments OFF or 0 hides the graticule grid.
ON or 1 shows the graticule grid.

Examples `DISPlay:BLUEtooth:FDVTime:WINDow:TRACe:GRATicule:GRID:STATe ON` specifies to show the graticule grid.

DISPlay:BLUEtooth:FDVTime:Y[:SCALe]:AUTO (No Query Form)

Rescales the vertical scale automatically to fit the waveform to the screen in the Bluetooth Frequency Deviation vs. Time display.

Conditions Measurement views: Frequency Deviation vs. Time

Group Display commands

Syntax
```
DISPlay:BLUEtooth:FDVTime:Y[:SCALe]:AUTO
```

Arguments None

Examples `DISPlay:BLUEtooth:FDVTime:Y:SCALe:AUTO` rescales the vertical scale automatically to fit the waveform to the screen.

DISPlay:BLUEtooth:FDVTime:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (center point of the vertical axis) in the Bluetooth Frequency Deviation vs. Time display.

Conditions Measurement views: Frequency Deviation vs. Time

Group Display commands

Syntax
```
DISPlay:BLUEtooth:FDVTime:Y[:SCALe]:OFFSet <value>
DISPlay:BLUEtooth:FDVTime:Y[:SCALe]:OFFSet?
```

Arguments <value> ::= <NRf> the vertical offset. Range: -60 MHz to +60 MHz.

Examples `DISPlay:BLUEtooth:FDVTime:Y:SCALe:OFFSet -14.5E+3` sets the vertical offset to -14.5 kHz.

DISPlay:BLUEtooth:MEASview:DELete (No Query Form)

Deletes the specified Bluetooth analysis view.

Conditions Measurement views: All.

Group Display commands

Syntax
```
DISPlay:BLUEtooth:MEASview:DELete
{CONS|SUMM|FDVT|EDI|BOBW|STAB|BIBE|DRIF}
```

Related Commands DISPlay:BLUEtooth:MEASview:NEW, DISPlay:BLUEtooth:MEASview:SELect

Arguments
CONS deletes the BT Constellation view.
SUMM deletes the BT Summary view.
FDVT deletes the BT Freq Dev vs. Time view.
EDI deletes the BT Eye Diagram view.
BOBW deletes the BT 20dB BW view.
STAB deletes the BT Symbol Table view.
BIBE deletes the BT InBand Emission view.
DRIF deletes the BT Drift Table view.

If you attempt to delete a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running" will be returned.

Examples `DISPLAY:BLUETOOTH:MEASVIEW:DELETE` CONS deletes the BT Constellation view.

DISPlay:BLUEtooth:MEASview:NEW (No Query Form)

Displays a new measurement view for Bluetooth measurements.

Conditions Measurement views: Bluetooth measurements.

Group Display commands

Syntax
```
DISPlay:BLUEtooth:MEASview:NEW
{CONS|SUMM|FDVT|EDI|BOBW|STAB|BIBE|DRIF}
```

Arguments
CONS creates a new BT Constellation view.
SUMM creates a new BT Summary view.
FDVT creates a new BT Freq Dev vs. Time view.
EDI creates a new BT Eye Diagram view.
BOBW creates a new BT 20dB BW view.
STAB creates a new BT Symbol Table view.
BIBE creates a new BT InBand Emission view.
DRIF creates a new BT Drift Table view.

If you attempt to open a view that is currently displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Examples `DISPLAY:BLUETOOTH:MEASVIEW:NEW` CONS creates a new Bluetooth Constellation view.

DISPlay:BLUEtooth:MEASview:SELect

Selects a measurement view on the screen in the Bluetooth measurements. The query command returns the currently selected view.

Conditions Measurement views: Bluetooth measurements.

Group Display commands

Syntax
```
DISPlay:BLUEtooth:MEASview:SELect
{CONS|SUMM|FDVT|EDI|BOBW|STAB|BIBE|DRIF}
DISPlay:BLUEtooth:MEASview:SELect?
```

Arguments

CONS selects the BT Constellation view.
SUMM selects the BT Summary view.
FDVT selects the BT Freq Dev vs. Time view.
EDI selects the BT Eye Diagram view.
BOBW selects the BT 20dB BW view.
STAB selects the BT Symbol Table view.
BIBE selects the BT InBand Emission view.
DRIF selects the BT Drift Table view.

If you attempt to selects a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running" will be returned.

Examples

`DISPLAY:BLUETOOTH:MEASVIEW:SELECT` EDI selects the Bluetooth Eye Diagram view.

DISPlay:BOBW:SELected:BANDwidth

Sets or queries the bandwidth to measure in the Bluetooth 20 dB BW view.

Conditions

Measurement views: Bluetooth 20 dB BW

Group

Display commands

Syntax

```
DISPlay:BOBW:SELected:BANDwidth { XDBM | XDB }
DISPlay:BOBW:SELected:BANDwidth?
```

Arguments

XDBM selects the x dB bandwith to measrure.
XDB selects the x dB bandwith to measrure.

Examples

`DISPlay:BOBW:SELected:BANDwidth XDB` selects the x dB bandwidth to measure.

DISPlay:CCDF:LEGend:STATe

This command does not apply to SignalVu-PC or connected instruments.

Determines whether to show or hide the trace legend in the CCDF view. The legend indicates the trace detection and function on the screen for each displayed CCDF trace.

Conditions

Measurement views: CCDF

Group Display commands

Syntax
```
DISPlay:CCDF:LEGend:STATe { OFF | ON | 0 | 1 }
DISPlay:CCDF:LEGend:STATe?
```

Arguments `OFF` or `0` hides the trace legend.

`ON` or `1` shows the trace legend.

Examples `DISPLAY:CCDF:LEGEND:STATEON` shows the trace legend on the screen.

DISPlay:CCDF:WINDow:TRACe:GRATicule:GRID:STATe

This command does not apply to SignalVu-PC or connected instruments.

Determines whether to show or hide the graticule grid on the screen.

Conditions Measurement views: CCDF

Group Display commands

Syntax
```
DISPlay:CCDF:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON
| 0 | 1 }
DISPlay:CCDF:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples `DISPLAY:CCDF:WINDOW:TRACE:GRATICULE:GRID:STATEON` shows the graticule grid on the screen in the CCDF view.

DISPlay:CONSte:MPHase

Sets or queries the multiplication constant of the phase multiplication constellation display for a CPM signal. This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to CPM.

Conditions Measurement views: Constellation

Group Display commands

Syntax
```
DISPlay:CONSte:MPHase { P1 | P2 | P4 | P8 | P16 | P32 }
DISPlay:CONSte:MPHase?
```

Arguments The following table shows the arguments and phase multiplier.

Phase multiplication

Argument	Phase multiplier
P1	1
P2	2
P4	4
P8	8
P16	16
P32	32

Examples `DISPLAY:CONSTE:MPHASE P8` sets the phase multiplier to 8.

DISPlay:CONSte:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions Measurement views: Constellation

Group Display commands

Syntax
```
DISPlay:CONSte:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON
| 0 | 1 }
DISPlay:CONSte:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples `DISPLAY:CONSTE:WINDOW:TRACE:GRATICULE:GRID:STATEON` shows the graticule grid on the screen.

DISPlay:DDEMod:MEASview:DELete (No Query Form)

Deletes the measurement view in the general purpose digital modulation measurements.

Conditions Measurement views: General purpose digital modulation

Group Display commands

Syntax

```
DISPlay:DDEMod:MEASview:DELete { CONSte | DIQVtime |
EDIagram | EVM | FDVTime | MERRor | PERRor | SIGNalqual |
STABle | TDIagram }
```

Arguments The following table lists the arguments. The arguments are the string type.

Table 2-22: Modulation measurement views

Argument	View
CONSte	Constellation
DIQVtime	Demodulated I&Q versus Time
EDIagram	Eye diagram
EVM	EVM (Error Vector Magnitude) versus Time
FDVTime	Frequency deviation versus Time
MERRor	Magnitude error versus Time
PERRor	Phase error versus Time
SIGNalqual	Signal quality
STABle	Symbol table
TDIagram	Trellis diagram

If you attempt to delete a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running" will be returned.

Examples `DISPLAY:DDEMOD:MEASVIEW:DELETECONSte` deletes the constellation view.

DISPlay:DDEMod:MEASview:NEW (No Query Form)

Displays a new measurement view in the general purpose digital modulation measurements.

Conditions Measurement views: General purpose digital modulation

Group Display commands

Syntax
```
DISPlay:DDEMod:MEASview:NEW { CONSte | DIQVtime | EDIagram
| EVM | FDVTime | MERRor | PERRor | SIGNalqual | STABle |
TDIagram }
```

Arguments (See Table 2-22 on page 2-444.) If you attempt to open a view that is currently displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Examples `DISPLAY:DDEMOD:MEASVIEW:NEWCONSte` creates the constellation view.

DISPlay:DDEMod:MEASview:SELect

Selects a measurement view in the general purpose digital modulation measurements on the screen. The query command returns the currently selected view.

Selecting a measurement optimizes it. Other measurements may be optimized as a side effect. Refer to the DISPlay:WINDow:OPTimized:MEASurement? query.

Conditions Measurement views: General purpose digital modulation

Group Display commands

Syntax
```
DISPlay:DDEMod:MEASview:SELect { CONSte | DIQVtime |
EDIagram | EVM | FDVTime | MERRor | PERRor | SIGNalqual |
STABle | TDIagram }
DISPlay:DDEMod:MEASview:SELect?
```

Arguments (See Table 2-22 on page 2-444.) If you attempt to select a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running") will be returned.

Examples `DISPLAY:DDEMOD:MEASVIEW:SELECTCONSte` selects the constellation view.

DISPlay:DDEMod:RADix

Sets or queries the base of symbols. This command is effective in the symbol table.

Conditions Measurement views: Symbol table

Group Display commands

Syntax
```
DISPlay:DDEMod:RADix { BINary | HEXadecimal }
DISPlay:DDEMod:RADix?
```

Arguments `BINary` selects binary notation.

`HEXadecimal` selects hexadecimal notation.

Examples `DISPLAY:DDEMOD:RADIXBINary` selects binary notation for the symbol table.

DISPlay:DDEMod:X[:SCALe]

Sets or queries the horizontal scale (full-scale time) for the time measurements in the general purpose digital modulation analysis. Programming a specified scale sets DISPlay:DDEMod:X[:SCALe]:AUTO:STATe OFF.

Conditions Measurement views: EVM versus Time, Magnitude error versus Time, Phase error versus Time

Group Display commands

Syntax
```
DISPlay:DDEMod:X[:SCALe] <value>
DISPlay:DDEMod:X[:SCALe]?
```

Related Commands DISPlay:DDEMod:X[:SCALe]:AUTO:STATe, DISPlay:DDEMod:X[:SCALe]:OFFSet

Arguments `<value>::=<NRf>` specifies the horizontal scale in full-scale time. Use the DISPlay:DDEMod:X[:SCALe]:MAXimum? and DISPlay:DDEMod:X[:SCALe]:MINimum? queries to get the upper and lower limits of the setting range. The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples `DISPLAY:DDEMOD:X:SCALE 1.5us` sets the horizontal scale to 1.5 µs.

DISPlay:DDEMod:X[:SCALe]:AUTO (No Query Form)

Sets the horizontal scale automatically to fit the waveform to the screen in the general purpose digital modulation analysis. Executing this command sets DISPlay:DDEMod:X[:SCALe]:AUTO:STATe ON.

Conditions Measurement views: EVM versus Time, Magnitude error versus Time, Phase error versus Time

Group Display commands

Syntax

```
DISPlay:DDEMod:X[:SCALe]:AUTO
```

Related Commands DISPlay:DDEMod:X[:SCALe]:AUTO:STATe

Arguments None

Examples `DISPLAY:DDEMOD:X:SCALE:AUTO` sets the horizontal scale automatically to fit the waveform to the screen.

DISPlay:DDEMod:X[:SCALe]:AUTO:STATe

Determines whether to set the horizontal scale automatically or manually.

Conditions Measurement views: EVM versus Time, Magnitude error versus Time, Phase error versus Time

Group Display commands

Syntax

```
DISPlay:DDEMod:X[:SCALe]:AUTO:STATe { OFF | ON | 0 | 1 }
DISPlay:DDEMod:X[:SCALe]:AUTO:STATe?
```

Arguments `OFF` or `0` specifies that the horizontal scale is set manually. To set it, use the DISPlay:DDEMod:X[:SCALe] and DISPlay:DDEMod:X[:SCALe]:OFFSet commands.

`ON` or `1` specifies that the horizontal scale is set automatically.

Examples `DISPLAY:DDEMOD:X:SCALE:AUTO:STATE ON` specifies that the horizontal scale is set automatically.

DISPlay:DDEMod:X[:SCALe]:MAXimum? (Query Only)

Queries the upper limit of the horizontal scale setting range.

Conditions Measurement views: EVM versus Time, Magnitude error versus Time, Phase error versus Time

Group Display commands

Syntax `DISPlay:DDEMod:X[:SCALe]:MAXimum?`

Arguments None

Returns `<NRf>` The upper limit of the horizontal scale setting range.
The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples `DISPLAY:DDEMOD:X:SCALE:MAXIMUM?` might return `18.135E-3`, indicating that the upper limit of the horizontal scale setting range is 18.135 ms.

DISPlay:DDEMod:X[:SCALe]:MINimum? (Query Only)

Queries the lower limit of the horizontal scale setting range.

Conditions Measurement views: EVM versus Time, Magnitude error versus Time, Phase error versus Time

Group Display commands

Syntax `DISPlay:DDEMod:X[:SCALe]:MINimum?`

Arguments None

Returns `<NRf>` The lower limit of the horizontal scale setting range.
The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples `DISPLAY:DDEMOD:X:SCALE:MINIMUM` might return `10.0E-9`, indicating that the lower limit of the horizontal scale setting range is 10.0 ns.

DISPlay:DDEMod:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (left edge) for the time measurements in the general purpose digital modulation analysis. Programming a specified offset sets DISPlay:DDEMod:X[:SCALe]:AUTO:STATe OFF.

Conditions Measurement views: EVM versus Time, Magnitude error versus Time, Phase error versus Time

Group Display commands

Syntax

```
DISPlay:DDEMod:X[:SCALe]:OFFSet <value>
DISPlay:DDEMod:X[:SCALe]:OFFSet?
```

Related Commands DISPlay:DDEMod:X[:SCALe]:AUTO:STATe, DISPlay:DDEMod:X[:SCALe]

Arguments `<value>::=<NRf>` specifies the minimum horizontal value.

Use the DISPlay:DDEMod:X[:SCALe]:OFFSet:MAXimum? and DISPlay:DDEMod:X[:SCALe]:OFFSet:MINimum? queries to get the upper and lower limits of the setting range.

The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples `DISPLAY:DDEMOD:X:SCALE:OFFSET 20.075us` sets the minimum horizontal value to 20.075 µs.

DISPlay:DDEMod:X[:SCALe]:OFFSet:MAXimum? (Query Only)

Queries the upper limit of the horizontal offset setting range.

Conditions Measurement views: EVM versus Time, Magnitude error versus Time, Phase error versus Time

Group Display commands

Syntax

```
DISPlay:DDEMod:X[:SCALe]:OFFSet:MAXimum?
```

Arguments None

Returns <NRf> The upper limit of the horizontal offset setting range. The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples DISPLAY:DDEMOD:X:SCALE:OFFSET:MAXIMUM? might return -1.812E-3, indicating that the upper limit of the horizontal offset setting range is -1.812 ms.

DISPlay:DDEMod:X[:SCALe]:OFFSet:MINimum? (Query Only)

Queries the lower limit of the horizontal offset setting range.

Conditions Measurement views: EVM versus Time, Magnitude error versus Time, Phase error versus Time

Group Display commands

Syntax DISPlay:DDEMod:X[:SCALe]:OFFSet:MINimum?

Arguments None

Returns <NRf> The lower limit of the horizontal offset setting range. The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples DISPLAY:DDEMOD:X:SCALE:OFFSET:MINIMUM? might return -16.28E-3, indicating that the lower limit of the horizontal offset setting range is -16.28 ms.

DISPlay:DDEMod:X[:SCALe]:RESet (No Query Form)

Presets the horizontal scale to the default value for the time measurements in the general purpose digital modulation analysis.

Conditions Measurement views: EVM versus Time, Magnitude error versus Time, Phase error versus Time

Group Display commands

Syntax `DISPlay:DDEMod:X[:SCALe]:RESet`

Arguments None

Examples `DISPLAY:DDEMOD:X:SCALE:RESET` presets the horizontal scale to the default value.

DISPlay:DIAGram:X[:SCALe]

Sets or queries the horizontal range for the eye and trellis diagrams in the general purpose digital modulation analysis.

Conditions Measurement views: Eye diagram, Trellis diagram

Group Display commands

Syntax
```
DISPlay:DIAGram:X[:SCALe] <value>
DISPlay:DIAGram:X[:SCALe]?
```

Arguments `<value>::=<NR1>` specifies the horizontal range.
Range: 1 to 16 symbols.

Examples `DISPLAY:DIAGRAM:X:SCALE 3` sets the horizontal range to 3 symbols in the eye and trellis diagrams.

DISPlay:DIAGram:X[:SCALe]:RESet (No Query Form)

Presets the horizontal scale to the default value for the eye and trellis diagrams in the general purpose digital modulation analysis.

Conditions Measurement views: Eye diagram, Trellis diagram

Group Display commands

Syntax `DISPlay:DIAGram:X[:SCALe]:RESet`

Arguments None

Examples `DISPLAY:DIAGRAM:X:SCALE:RESET` presets the horizontal scale to the default value for the eye and trellis diagrams.

DISPlay:DIQVtime:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions Measurement views: Demodulated I&Q versus Time

Group Display commands

Syntax

```
DISPlay:DIQVtime:WINDow:TRACe:GRATicule:GRID:STATe { OFF |
ON | 0 | 1 }
DISPlay:DIQVtime:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples `DISPLAY:DIQVTIME:WINDOW:TRACE:GRATICULE:GRID:STATEON` shows the graticule grid on the screen in the Demodulated I&Q versus Time view.

DISPlay:DIQVtime:Y[:SCALe]

Sets or queries the vertical range of the Demodulated I&Q versus Time graph.

Conditions Measurement views: Demodulated I&Q versus Time

Group Display commands

Syntax

```
DISPlay:DIQVtime:Y[:SCALe] <value>
DISPlay:DIQVtime:Y[:SCALe]?
```

Arguments `<value>::=<NRf>` specifies the vertical range. Range: 1 µV to 10 V.

Examples `DISPLAY:DIQVTIME:Y:SCALE 2.5` sets the vertical range to 2.5 V in the Demodulated I&Q versus Time graph.

DISPlay:DIQVtime:Y[:SCALe]:AUTO (No Query Form)

Sets the vertical scale automatically to fit the waveform to the screen in the Demodulated I&Q versus Time display.

Conditions Measurement views: Demodulated I&Q versus Time

Group Display commands

Syntax `DISPlay:DIQVtime:Y[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:DIQVTIME:Y:SCALE:AUTO` sets the vertical scale automatically to fit the waveform to the screen.

DISPlay:DIQVtime:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (center point of the vertical axis) of the Demodulated I&Q versus Time graph.

Conditions Measurement views: Demodulated I&Q versus Time

Group Display commands

Syntax
```
DISPlay:DIQVtime:Y[:SCALe]:OFFSet <value>
DISPlay:DIQVtime:Y[:SCALe]:OFFSet?
```

Arguments `<value>::=<NRf>` specifies the vertical offset. Range: -5 to +5 V.

Examples `DISPLAY:DIQVTIME:Y:SCALE:OFFSET -0.5` sets the vertical offset to -0.5 V in the Demodulated I&Q versus Time graph.

DISPlay:DPX:DGRam:TIME[:SCALe]:OFFSet:DIVision

Sets or queries the DPXogram vertical time offset in divisions.

Conditions Measurement views: DPX spectrum

Group Display commands

Syntax
```
DISPlay:DPX:DGRam:TIME[:SCALe]:OFFSet:DIVision <value>
DISPlay:DPX:DGRam:TIME[:SCALe]:OFFSet:DIVision?
```

Arguments <value> ::= <NRf> vertical offset value in divisions.

Examples `DISPLAY:DPX:DGRAM:TIME:SCALE:OFFSET:DIVISION 10` sets the veritcal time off set to 10 divisions.

DISPlay:DPX:DGRam:TIME[:SCALe]:PDIVision

Sets or queries the DPXogram vertical time scale per division.

Conditions MMeasurement views: DPX spectrum

Group Display commands

Syntax
```
DISPlay:DPX:DGRam:TIME[:SCALe]:PDIVision <value>
DISPlay:DPX:DGRam:TIME[:SCALe]:PDIVision?
```

Arguments <value> :: <NRf> vertical value in divisions.

Examples `DISPLAY:DPX:DGRAM:TIME:SCALE:PDIVISION 10` sets the vertical time scale to 10 seconds per division.

DISPlay:DPX:DGRam:TIME[:SCALe]:RESet (No Query Form)

Presets the time scale and offset to the default value for the DPXogram measurement.

Conditions Measurement views: DPX spectrum

Group Display commands

Syntax `DISPlay:DPX:DGRam:TIME[:SCALe]:RESet`

Arguments None

Examples `DISPLAY:DPX:DGRAM:TIME:SCALE:RESET` presets the time scale and offset to the default value.

DISPlay:DPX:DGRam:TSTamp:STATe

Determines whether to display the time stamp readout in the DPXogram display.

Conditions Measurement views: DPX spectrum

Group Display commands

Syntax
```
DISPlay:DPX:DGRam:TSTamp:STATe { OFF | ON | 0 | 1 }
DISPlay:DPX:DGRam:TSTamp:STATe?
```

Arguments `OFF` or `0` hides the time stamp readout.

`ON` or `1` shows the time stamp readout.

Examples `DISPLAY:DPX:DGRAM:TSTAMP:STATE 0` hides the time stamp readout in the DPXogram display.

DISPlay:DPX:DGRam:Y[:SCALe]:AUTO (No Query Form)

Rescales the height axis automatically to fit the waveform to the screen in the DPXogram display.

Conditions Measurement views: DPX spectrum

Group Display commands

Syntax `DISPlay:DPX:DGRam:Y[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:DPX:DGRAM:Y:AUTO` rescales the height scale automatically to fit the waveform to the DPXogram display.

DISPlay:DPX:DGRam:Y[:SCALe]:RESet (No Query Form)

Resets the height scale of the DPXogram display to the default values:

Height position = 0 dB and Height scale = 100 dB.

Conditions Measurement views: DPX spectrum

Group Display commands

Syntax `DISPlay:DPX:DGRam:Y[:SCALe]:RESet`

Arguments None

Examples `DISPLAY:DPX:DGRAM:Y:RESET` resets the height scale of the DPXogram display.

DISPlay:DPX:LEGend:STATe

Determines whether to show or hide the trace legend on the display. The legend indicates the trace detection and function on the screen for each displayed trace.

Conditions Measurement views: DPX spectrum

Group Display commands

Syntax

```
DISPlay:DPX:LEGend:STATe { OFF | ON | 0 | 1 }
DISPlay:DPX:LEGend:STATe?
```

Arguments `OFF` or `0` hides the trace legend.

`ON` or `1` shows the trace legend.

Examples `DISPLAY:DPX:LEGEND:STATE ON` shows the trace legend on the screen.

DISPlay:DPX:PHASe:Y[:SCALe]:AXIS

Sets or queries the vertical axis representation.

Conditions Measurement views: DPX Phase

Group Display commands

Syntax

```
DISPlay:DPX:PHASe:Y[:SCALe]:AXIS { MODulopi | CONTinuous }
DISPlay:DPX:PHASe:Y[:SCALe]:AXIS?
```

Arguments `MODulopi` (modulo π) shows the phase constrained within ±180° along the vertical axis.

`CONTinuous` shows the phase as continuous quantity along the vertical axis.

Examples `DISPLAY:DPX:PHASE:Y[:SCALE]:AXIS` MODulopi selects modulo π representation for the vertical axis.

DISPlay:DPX:PHASe:Y[:SCALe]:AXIS:OFFSet

Sets or queries the vertical offset in the DPX Phase view.

Conditions Measurement views: DPX Phase

Group Display commands

Syntax

```
DISPlay:DPX:PHASe:Y[:SCALe]:AXIS:OFFSet <value>
DISPlay:DPX:PHASe:Y[:SCALe]:AXIS:OFFSet?
```

Arguments <value> :: <NRf> specifies the vertical offset. Range: –180° to +180°.

Examples `DISPLAY:DPX:PHASE:Y[:SCALE]:AXIS:OFFSET` –158.5 sets the vertical offset to –158.5° in the view.

DISPlay:DPX:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions Measurement views: DPX spectrum

Group Display commands

Syntax

```
DISPlay:DPX:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON |
0 | 1 }
DISPlay:DPX:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples `DISPLAY:DPX:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the screen in the DPX spectrum view.

DISPlay:DPX:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions Measurement views: DPX spectrum

Group Display commands

Syntax
```
DISPlay:DPX:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:DPX:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples `DISPLAY:DPX:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the screen in the DPX spectrum view.

DISPlay:DPX:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the top edge of the vertical axis) of the DPX spectrum view.

Conditions Measurement views: DPX spectrum

Group Display commands

Syntax
```
DISPlay:DPX:Y[:SCALe]:OFFSet <value>
DISPlay:DPX:Y[:SCALe]:OFFSet?
```

Arguments `<value> ::= <NRf>` specifies the vertical offset. Range: -270 to +150 dBm. The amplitude unit is set by the [SENSe]:POWer:UNITs command (default: dBm).

You can omit the unit in the argument. When you include the unit in the argument, only dBm is allowed. For the other amplitude units, omit the unit in the argument.

Examples `DISPLAY:DPX:Y[:SCALE]:OFFSET -12.5dBm` sets the vertical offset to -12.5 dBm.

DISPlay:DPX:Y[:SCALe]:PDIVision

Sets or queries the vertical scale (per division) in the DPX spectrum view.

Conditions Measurement views: DPX spectrum

Group Display commands

Syntax
```
DISPlay:DPX:Y[:SCALe]:PDIVision <value>
DISPlay:DPX:Y[:SCALe]:PDIVision?
```

Related Commands [SENSe]:POWer:UNITs

Arguments <value> :: <NRf> specifies the vertical scale (per division).
Range: 2 to 20 dB/div.

Examples DISPLAY:DPX:Y[:SCALE]:PDIVISION0.5 sets the vertical scale to 0.5 dB/div.

DISPlay:EDIagram:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen in the eye diagram.

Conditions Measurement views: Eye diagram

Group Display commands

Syntax
```
DISPlay:EDIagram:WINDow:TRACe:GRATicule:GRID:STATe { OFF |
ON | 0 | 1 }
DISPlay:EDIagram:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments OFF or 0 hides the graticule grid.

ON or 1 shows the graticule grid.

Examples DISPLAY:EDIAGRAM:WINDOW:TRACE:GRATICULE:GRID:STATE ON shows the graticule grid on the screen in the eye diagram.

DISPlay:EDIagram:Y[:SCALe]

Sets or queries the vertical range of the eye diagram.

Conditions Measurement views: Eye diagram

Group Display commands

Syntax
```
DISPlay:EDIagram:Y[:SCALe] <value>
DISPlay:EDIagram:Y[:SCALe]?
```

Arguments `<value> ::` `<NRf>` specifies the vertical range. Range: 1 µ to 100 (unitless).

Examples `DISPLAY:EDIAGRAM:Y:SCALE 2.5` sets the vertical range to 2.5 in the eye diagram.

DISPlay:EDIagram:Y[:SCALe]:AUTO (No Query Form)

Sets the vertical scale automatically to fit the waveform to the screen in the eye diagram.

Conditions Measurement views: Eye diagram

Group Display commands

Syntax `DISPlay:EDIagram:Y[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:EDIAGRAM:Y:SCALE:AUTO` sets the vertical scale automatically to fit the waveform to the screen.

DISPlay:EDIagram:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (center point of the vertical axis) of the eye diagram.

Conditions Measurement views: Eye diagram

Group Display commands

Syntax

```
DISPlay:EDIagram:Y[:SCALe]:OFFSet <value>
DISPlay:EDIagram:Y[:SCALe]:OFFSet?
```

Arguments

<value> :: <NRf> specifies the minimum vertical value.
Range: -50 to +50 (unitless).

Examples

`DISPLAY:EDIAGRAM:Y:SCALE:OFFSET -0.5` sets the vertical offset to -0.5 in the eye diagram.

DISPlay:EVM:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries the graticule grid view state for the EVM view.

Conditions

Measurement views: EVM versus Time

Group

Display commands

Syntax

```
DISPlay:EVM:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON |
0 | 1 }
DISPlay:EVM:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples

`DISPLAY:EVM:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the EVM view.

DISPlay:EVM:Y[:SCALe]

Sets or queries the vertical range of the EVM versus Time graph.

Conditions

Measurement views: EVM versus Time

Group

Display commands

Syntax

```
DISPlay:EVM:Y[:SCALe] <value>
DISPlay:EVM:Y[:SCALe]?
```

Related Commands DISPlay:EVM:Y[:SCALe]:OFFSet

Arguments `<value> ::` `<NRf>` specifies the vertical range. Range: 1 to 100%.

Examples `DISPLAY:EVM:Y:SCALE 50` sets the vertical range to 50% in the EVM versus Time graph.

DISPlay:EVM:Y[:SCALe]:AUTO (No Query Form)

Sets the vertical scale automatically to fit the waveform to the screen in the EVM versus Time display.

Conditions Measurement views: EVM versus Time

Group Display commands

Syntax `DISPlay:EVM:Y[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:EVM:Y:SCALE:AUTO` sets the vertical scale automatically to fit the waveform to the screen.

DISPlay:EVM:Y[:SCALe]:OFFSet

Sets or queries the minimum vertical value (bottom edge) of the EVM versus Time graph.

Conditions Measurement views: EVM versus Time

Group Display commands

Syntax
```
DISPlay:EVM:Y[:SCALe]:OFFSet <value>
DISPlay:EVM:Y[:SCALe]:OFFSet?
```

Related Commands DISPlay:EVM:Y[:SCALe]

Arguments `<value> ::` `<NRf>` specifies the minimum vertical value. Range: -100 to 100%.

Examples `DISPLAY:EVM:Y:SCALE:OFFSET -9.5` sets the minimum vertical value to -9.5% in the EVM versus Time graph.

DISPlay:FDVTime:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions Measurement views: Frequency deviation versus Time

Group Display commands

Syntax
```
DISPlay:FDVTime:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:FDVTime:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples `DISPLAY:FDVTIME:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the Frequency deviation versus Time view.

DISPlay:FDVTime:Y[:SCALe]

Sets or queries the vertical range of the Frequency deviation versus Time graph.

Conditions Measurement views: Frequency deviation versus Time

Group Display commands

Syntax
```
DISPlay:FDVTime:Y[:SCALe] <value>
DISPlay:FDVTime:Y[:SCALe]?
```

Arguments `<value> ::` `<NRf>` specifies the vertical range. Range: 10 Hz to 120 MHz.

Examples `DISPLAY:FDVTIME:Y:SCALE 30MHz` sets the vertical range to 30 MHz in the Frequency deviation versus Time graph.

DISPlay:FDVTime:Y[:SCALe]:AUTO (No Query Form)

Sets the vertical scale automatically to fit the waveform to the screen in the Frequency deviation versus Time view.

Conditions Measurement views: Frequency deviation versus Time

Group Display commands

Syntax `DISPlay:FDVTime:Y[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:FDVTIME:Y:SCALE:AUTO` sets the vertical scale automatically to fit the waveform to the screen.

DISPlay:FDVTime:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (center point of the vertical axis) in the Frequency deviation versus Time graph.

Conditions Measurement views: Frequency deviation versus Time

Group Display commands

Syntax
```
DISPlay:FDVTime:Y[:SCALe]:OFFSet <value>
DISPlay:FDVTime:Y[:SCALe]:OFFSet?
```

Arguments `<value> ::` `<NRf>` specifies the vertical offset. Range: -60 MHz to +60 MHz.

Examples `DISPLAY:FDVTIME:Y:SCALE:OFFSET -14.5MHz` sets the vertical offset to -14.5 MHz in the Frequency deviation versus Time graph.

DISPlay:{FSETtling|PSETtling}:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the Frequency or Phase Settling views.

Conditions Measurement views: Frequency and Phase Settling

Group Display commands

Syntax

```
DISPlay:{FSETtling|PSETtling}:MARKer:SHOW:STATe { OFF | ON
| 0 | 1 }
DISPlay:{FSETtling|PSETtling}:MARKer:SHOW:STATe?
```

Arguments `OFF` or `0` hides the readout for the selected marker in the view.

`ON` or `1` shows the readout for the selected marker in the view.

Examples `DISPLAY:FSETTLING:MARKER:SHOW:STATE ON` shows the readout for the selected marker in the view.

DISPlay:{FSETtling|PSETtling}:TIME:DECimal

Sets or queries the number of values to the right of the decimal point to include in the Settling Time and Settling Time from Trigger results.

Conditions Measurement views: Frequency and Phase Settling Time

Group Display commands

Syntax

```
DISPlay:{FSETtling|PSETtling}:TIME:DECimal <value>
DISPlay:{FSETtling|PSETtling}:TIME:DECimal?
```

Arguments `<value> ::= <NR1>` specifies the number digits to the right of the decimal point to include in Settling Time and Settling Time from Trigger results. Range: 0 to 6.

Examples `DISPLAY:PSETTLING:TIME:DECIMAL 3` sets the number of values to the right of the decimal point to 3 for the Phase Settling time measurement.

DISPlay:{FSETtling|PSETtling}:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions Measurement views: Frequency and Phase Settling

Group Display commands

Syntax

```
DISPlay:{FSETtling|PSETtling}:WINDow:TRACe:GRATicule:GRID:
STATe { OFF | ON | 0 | 1 }
DISPlay:{FSETtling|PSETtling}:WINDow:TRACe:GRATicule:GRID:
STATe?
```

Arguments `OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples `DISPLAY:FSETTLING:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the screen for the Frequency Settling display.

DISPlay:{FSETtling|PSETtling}:X[:SCALe]

Sets or queries the horizontal scale (full-scale time) of the Settling Time graph. Programming a specified scale sets DISPlay:{FSETtling|PSETtling}:X[:SCALe] AUTO:STATe OFF.

Conditions Measurement views: Frequency versus Time

Group Display commands

Syntax

```
DISPlay:{FSETtling|PSETtling}:X[:SCALe] <value>
DISPlay:{FSETtling|PSETtling}:X[:SCALe]?
```

Related Commands DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO:STATeDISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet,

Arguments `<value>::={ <NRf> | MAXimum | MINimum }` specifies the horizontal scale in full-scale time. `MAXimum` and `MINimum` represent the upper and lower limits of the setting range, respectively.

Use the DISPlay:{FSETtling|PSETtling}:X[:SCALe]:MAXimum? and DISPlay:{FSETtling|PSETtling}:X[:SCALe]:MINimum? queries to get the upper and lower limit values of the setting range.

Examples `DISPLAY:FSETTLING:X:SCALE 25.6us` sets the horizontal scale to 25.6 µs.

DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO (No Query Form)

Sets the horizontal scale automatically to fit the waveform to the screen in the Frequency and Phase Settling views. Executing this command sets DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO:STATe ON.

Conditions Measurement views: Frequency and Phase Settling

Group Display commands

Syntax
```
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO
```

Related Commands DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO:STATe

Arguments None

Examples `DISPLAY:FSETTLING:X:SCALE:AUTO` sets the horizontal scale automatically to fit the waveform to the screen.

DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO:STATe

Determines whether to set the horizontal scale automatically or manually.

Conditions Measurement views: Frequency and Phase Settling

Group Display commands

Syntax
```
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO:STATe { OFF |
ON | 0 | 1 }
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO:STATe?
```

Arguments

`OFF` or `0` specifies that the horizontal scale is set manually. To set it, use the DISPlay:{FSETtling|PSETtling}:X[:SCALe] and DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet commands.

`ON` or `1` specifies that the horizontal scale is set automatically.

Examples

`DISPLAY:FSETTLING:X:SCALE:AUTO:STATE ON` specifies that the horizontal scale is set automatically.

DISPlay:{FSETtling|PSETtling}:X[:SCALe]:MAXimum? (Query Only)

Queries the upper limit of the horizontal scale setting range.

Conditions

Measurement views: Frequency and Phase Settling

Group

Display commands

Syntax

`DISPlay:{FSETtling|PSETtling}:X[:SCALe]:MAXimum?`

Related Commands

DISPlay:{FSETtling|PSETtling}:X[:SCALe]

Arguments

None

Returns

`<NRf>` The upper limit of the horizontal scale setting range.

Examples

`DISPLAY:FSETTLING:X:SCALE:MAXIMUM?` might return `18.135E-3`, indicating that the upper limit of the horizontal scale setting range is 18.135 ms.

DISPlay:{FSETtling|PSETtling}:X[:SCALe]:MINimum? (Query Only)

Queries the lower limit of the horizontal scale setting range.

Conditions

Measurement views: Frequency and Phase Settling

Group

Display commands

Syntax

`DISPlay:{FSETtling|PSETtling}:X[:SCALe]:MINimum?`

Related Commands DISPlay:FVTime:X[:SCALe]

Arguments None

Returns <NRf> The lower limit of the horizontal scale setting range.

Examples DISPLAY:FSETTLING:X:SCALE:MINIMUM? might return 10.0E-9, indicating that the lower limit of the horizontal scale setting range is 10.0 ns.

DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (left edge) of the Frequency versus Time graph. Programming a specified offset sets DISPlay:{FSETtling|PSETtling}:X[:SCALe] AUTO:STATe OFF.

Conditions Measurement views: Frequency and Phase Settling

Group Display commands

Syntax
```
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet <value>
DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet?
```

Related Commands DISPlay:{FSETtling|PSETtling}:X[:SCALe]:AUTO:STATe, DISPlay:{FSETtling|PSETtling}:X[:SCALe]

Arguments <value>::={ <NRf> | MAXimum | MINimum } specifies the horizontal offset. MAXimum and MINimum represent the upper and lower limits of the setting range, respectively.

Use the DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet:MAXimum? and DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet:MINimum? queries to get the upper and lower limit values of the setting range.

Examples DISPLAY:FSETTLING:X:SCALE:OFFSET 800ns sets the minimum horizontal value to 800 ns in the Frequency Settling graph.

DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet:MAXimum? (Query Only)

Queries the upper limit of the horizontal offset setting range.

Conditions Measurement views: Frequency and Phase Settling

Group Display commands

Syntax `DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet:MAXimum?`

Related Commands DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet

Arguments None

Returns <NRf> The upper limit of the horizontal offset setting range.

Examples `DISPLAY:FSETTLING:X:SCALE:OFFSET:MAXIMUM?` might return `338.186000E-6`, indicating that the upper limit of the horizontal offset setting range is 338.186 µs.

DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet:MINimum? (Query Only)

Queries the lower limit of the horizontal offset setting range.

Conditions Measurement views: Frequency and Phase Settling

Group Display commands

Syntax `DISPlay:{FSETtling|PSETtling}:X[:SCALe]:OFFSet:MINimum?`

Related Commands DISPlay:FVTime:X[:SCALe]:OFFSet

Arguments None

Returns <NRf> The lower limit of the horizontal offset setting range.

Examples `DISPLAY:FSETTLING:X:SCALE:OFFSET:MINIMUM?` might return `-338.186000E-6`, indicating that the lower limit of the horizontal offset setting range is -338.186 µs.

DISPlay:{FSETtling|PSETtling}:Y[:SCALe]

Sets or queries the vertical range of the Frequency versus Time graph.

Conditions Measurement views: Frequency and Phase Settling

Group Display commands

Syntax
```
DISPlay:{FSETtling|PSETtling}:Y[:SCALe] <value>
DISPlay:{FSETtling|PSETtling}:Y[:SCALe]?
```

Related Commands DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:OFFSet

Arguments `<value> ::  <NRf>` specifies the vertical range. Range: 10 Hz to 120 MHz.

Examples `DISPlay:FSETTLING:Y:SCALe 30E6` sets the vertical range to 30 MHz in the Frequency Settling graph.

DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:AUTO (No Query Form)

Sets the vertical scale automatically to fit the waveform to the screen in the Frequency and Phase Settling views.

Conditions Measurement views: Frequency and Phase Settling

Group Display commands

Syntax
```
DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:AUTO
```

Arguments None

Examples `DISPLAY:FSETTLING:Y:SCALE:AUTO` sets the vertical scale automatically to fit the waveform to the screen.

DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the center of the vertical axis) in the Frequency and Phase Settling graphs.

Conditions Measurement views: Frequency and Phase Settling

Group Display commands

Syntax

```
DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:OFFSet <value>
DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:OFFSet?
```

Related Commands DISPlay:{FSETtling|PSETtling}:Y[:SCALe]

Arguments `<value> ::` `<NRf>` specifies the vertical offset. Range: -60 MHz to +60 MHz.

Examples `DISPLAY:FVTIME:Y:SCALE:OFFSET -14.5E6` sets the vertical offset to -14.5 MHz in the Frequency Settling graph.

DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:PDIVision

Sets or queries the vertical scale (per division) of the Frequency and Phase Settling graphs.

Conditions Measurement views: Frequency and Phase Settling

Group Display commands

Syntax

```
DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:PDIVision <value>
DISPlay:{FSETtling|PSETtling}:Y[:SCALe]:PDIVision?
```

Arguments `<value> ::` `<NRf>` specifies the vertical scale (per division).
Range:

- Frequency Settling (Hz/div) : 500 to the Maximum bandwidth of instrument
- Phase Settling (degrees/div): 0.10 to X.

Examples `DISPLAY:PSETTLING:Y:SCALE:PDIVISION 2.7` sets the vertical scale to 27 degrees.

DISPlay:FVTime:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions Measurement views: Frequency versus Time

Group Display commands

Syntax

```
DISPlay:FVTime:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON
| 0 | 1 }
DISPlay:FVTime:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples `DISPLAY:FVTIME:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the Frequency versus Time view.

DISPlay:FVTime:X[:SCALe]

Sets or queries the horizontal scale (full-scale time) of the Frequency versus Time graph. Programming a specified scale sets DISPlay:FVTime:X[:SCALe] AUTO:STATe OFF.

Conditions Measurement views: Frequency versus Time

Group Display commands

Syntax

```
DISPlay:FVTime:X[:SCALe] <value>
DISPlay:FVTime:X[:SCALe]?
```

Related Commands DISPlay:FVTime:X[:SCALe]:AUTO:STATe, DISPlay:FVTime:X[:SCALe]:OFFSet

Arguments `<value>::={ <NRf> | MAXimum | MINimum }` specifies the horizontal scale in full-scale time. `MAXimum` and `MINimum` represent the upper and lower limits of the setting range, respectively.

Use the DISPlay:FVTime:X[:SCALe]:MAXimum? and DISPlay:FVTime:X[:SCALe]:MINimum? queries to get the upper and lower limit values of the setting range.

Examples `DISPLAY:FVTIME:X:SCALE 25.6us` sets the horizontal scale to 25.6 µs.

DISPlay:FVTime:X[:SCALe]:AUTO (No Query Form)

Sets the horizontal scale automatically to fit the waveform to the screen in the Frequency versus Time view. Executing this command sets DISPlay:FVTime X[:SCALe]:AUTO:STATe ON.

Conditions Measurement views: Frequency versus Time

Group Display commands

Syntax `DISPlay:FVTime:X[:SCALe]:AUTO`

Related Commands DISPlay:FVTime:X[:SCALe]:AUTO:STATe

Arguments None

Examples `DISPLAY:FVTIME:X:SCALE:AUTO` sets the horizontal scale automatically to fit the waveform to the screen.

DISPlay:FVTime:X[:SCALe]:AUTO:STATe

Determines whether to set the horizontal scale automatically or manually.

Conditions Measurement views: Frequency versus Time

Group Display commands

Syntax

```
DISPlay:FVTime:X[:SCALe]:AUTO:STATe { OFF | ON | 0 | 1 }
DISPlay:FVTime:X[:SCALe]:AUTO:STATe?
```

Arguments

`OFF` or `0` specifies that the horizontal scale is set manually. To set it, use the DISPlay:FVTime:X[:SCALe] and DISPlay:FVTime:X[:SCALe]:OFFSet commands.

`ON` or `1` specifies that the horizontal scale is set automatically.

Examples

`DISPLAY:FVTIME:X:SCALE:AUTO:STATE ON` specifies that the horizontal scale is set automatically.

DISPlay:FVTime:X[:SCALe]:MAXimum? (Query Only)

Queries the upper limit of the horizontal scale setting range.

Conditions

Measurement views: Frequency versus Time

Group

Display commands

Syntax

```
DISPlay:FVTime:X[:SCALe]:MAXimum?
```

Related Commands

DISPlay:FVTime:X[:SCALe]

Arguments

None

Returns

`<NRf>` The upper limit of the horizontal scale setting range.

Examples

`DISPLAY:FVTIME:X:SCALE:MAXIMUM?` might return `18.135E-3`, indicating that the upper limit of the horizontal scale setting range is 18.135 ms.

DISPlay:FVTime:X[:SCALe]:MINimum? (Query Only)

Queries the lower limit of the horizontal scale setting range.

Conditions

Measurement views: Frequency versus Time

Group

Display commands

Syntax `DISPlay:FVTime:X[:SCALe]:MINimum?`

Related Commands DISPlay:FVTime:X[:SCALe]

Arguments None

Returns `<NRf>` The lower limit of the horizontal scale setting range.

Examples `DISPLAY:FVTIME:X:SCALE:MINIMUM?` might return `10.0E-9`, indicating that the lower limit of the horizontal scale setting range is 10.0 ns.

DISPlay:FVTime:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (left edge) of the Frequency versus Time graph. Programming a specified offset sets DISPlay:FVTime:X[:SCALe] AUTO:STATe OFF.

Conditions Measurement views: Frequency versus Time

Group Display commands

Syntax
```
DISPlay:FVTime:X[:SCALe]:OFFSet <value>
DISPlay:FVTime:X[:SCALe]:OFFSet?
```

Related Commands DISPlay:FVTime:X[:SCALe]:AUTO:STATe, DISPlay:FVTime:X[:SCALe]

Arguments `<value>::={ <NRf> | MAXimum | MINimum }` specifies the horizontal offset. `MAXimum` and `MINimum` represent the upper and lower limits of the setting range, respectively.

Use the DISPlay:FVTime:X[:SCALe]:OFFSet:MAXimum? and DISPlay:FVTime:X[:SCALe]:OFFSet:MINimum? queries to get the upper and lower limit values of the setting range.

Examples `DISPLAY:FVTIME:X:SCALE:OFFSET 800ns` sets the minimum horizontal value to 800 ns in the Frequency versus Time graph.

DISPlay:FVTime:X[:SCALe]:OFFSet:MAXimum? (Query Only)

Queries the upper limit of the horizontal offset setting range.

Conditions Measurement views: Frequency versus Time

Group Display commands

Syntax `DISPlay:FVTime:X[:SCALe]:OFFSet:MAXimum?`

Related Commands DISPlay:FVTime:X[:SCALe]:OFFSet

Arguments None

Returns `<NRf>` The upper limit of the horizontal offset setting range.

Examples `DISPLAY:FVTIME:X:SCALE:OFFSET:MAXIMUM?` might return `-1.812E-3`, indicating that the upper limit of the horizontal offset setting range is -1.812 ms.

DISPlay:FVTime:X[:SCALe]:OFFSet:MINimum? (Query Only)

Queries the lower limit of the horizontal offset setting range.

Conditions Measurement views: Frequency versus Time

Group Display commands

Syntax `DISPlay:FVTime:X[:SCALe]:OFFSet:MINimum?`

Related Commands DISPlay:FVTime:X[:SCALe]:OFFSet

Arguments None

Returns `<NRf>` The lower limit of the horizontal offset setting range.

Examples `DISPLAY:FVTIME:X:SCALE:OFFSET:MINIMUM?` might return `-16.28E-3`, indicating that the lower limit of the horizontal offset setting range is -16.28 ms.

DISPlay:FVTime:Y[:SCALe]

Sets or queries the vertical range of the Frequency versus Time graph.

Conditions Measurement views: Frequency versus Time

Group Display commands

Syntax
```
DISPlay:FVTime:Y[:SCALe] <value>
DISPlay:FVTime:Y[:SCALe]?
```

Related Commands DISPlay:FVTime:Y[:SCALe]:OFFSet

Arguments `<value> ::= <NRf>` specifies the vertical range. Range: 10 Hz to 120 MHz.

Examples `DISPlay:FVTime:Y:SCALe 30MHz` sets the vertical range to 30 MHz in the Frequency versus Time graph.

DISPlay:FVTime:Y[:SCALe]:AUTO (No Query Form)

Sets the vertical scale automatically to fit the waveform to the screen in the Frequency versus Time view.

Conditions Measurement views: Frequency versus Time

Group Display commands

Syntax `DISPlay:FVTime:Y[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:FVTIME:Y:SCALE:AUTO` sets the vertical scale automatically to fit the waveform to the screen.

DISPlay:FVTime:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the center of the vertical axis) in the Frequency versus Time graph.

Conditions Measurement views: Frequency versus Time

Group Display commands

Syntax

```
DISPlay:FVTime:Y[:SCALe]:OFFSet <value>
DISPlay:FVTime:Y[:SCALe]:OFFSet?
```

Related Commands DISPlay:FVTime:Y[:SCALe]

Arguments `<value> ::` `<NRf>` specifies the vertical offset. Range: -60 MHz to +60 MHz.

Examples `DISPLAY:FVTIME:Y:SCALE:OFFSET -14.5MHz` sets the vertical offset to -14.5 MHz in the Frequency versus Time graph.

DISPlay:GENeral:MEASview:DELete (No Query Form)

Deletes a measurement view in the general signal viewing.

Conditions Measurement views: General signal viewing

Group Display commands

Syntax

```
DISPlay:GENeral:MEASview:DELete { SPECtrum | DPX | AVTime|
FVTime | PHVTime | IQVTime | SGRam | TOVerview }
```

Arguments The following table shows the arguments. The arguments are the string type.

Table 2-23: General signal viewing views

Argument	View
SPECtrum	Spectrum
DPX	DPX (Digital Phosphor) spectrum
AVTime	Amplitude versus Time
FVTime	Frequency versus Time

Table 2-23: General signal viewing views (cont.)

Argument	View
PHVTime	Phase versus Time
IQVTime	IQ versus Time
SGRam	Spectrogram
TOVerview	Time overview

If you attempt to delete a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running") will be returned.

Examples `DISPLAY:GENERAL:MEASVIEW:DELETEDPX` deletes the DPX spectrum view.

DISPlay:GENeral:MEASview:NEW (No Query Form)

Displays a new measurement view in the general signal viewing.

Conditions Measurement views: General signal viewing

Group Display commands

Syntax
```
DISPlay:GENeral:MEASview:NEW { SPECtrum | DPX | AVTime|
FVTime | PHVTime | IQVTime | SGRam | TOVerview }
```

Arguments (See Table 2-23 on page 2-480.) If you attempt to open a view that is currently displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Examples `DISPLAY:GENERAL:MEASVIEW:NEWDPX` creates the DPX spectrum view.

DISPlay:GENeral:MEASview:SELect

Selects a measurement view in the general signal viewing on the screen. The query command returns the currently selected view.

Selecting a measurement optimizes it. Other measurements may be optimized as a side effect. Refer to the DISPlay:WINDow:OPTimized:MEASurement? query.

Conditions Measurement views: General signal viewing

Group Display commands

Syntax
```
DISPlay:GENeral:MEASview:SELect { SPECtrum | DPX | AVTime|
FVTime | PHVTime | IQVTime | SGRam | TOVerview }
DISPlay:GENeral:MEASview:SELect?
```

Arguments (See Table 2-23 on page 2-480.) If you attempt to select a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running") will be returned.

The time overview (TOVerview) cannot be selected as the primary measurement. If you select it, the error (-200, "Execution error; Time Overview cannot be Primary measurement") will be returned. If you use the DISPlay:GENeral MEASview:SELect? query with the time overview as the only measurement active, the error (-200, "Execution error; Analysis selected is not running") will be returned.

Examples `DISPLAY:GENERAL:MEASVIEW:SELECTDPX` selects the DPX spectrum view.

DISPlay:GPRF:MEASview:DELete (No Query Form)

Deletes a selected measurement view in the RF measurements.

Conditions Measurement views: RF measurements

Group Display commands

Syntax
```
DISPlay:GPRF:MEASview:DELete { AM | FM | PM | CCDF | ACPower
| FSETtling | MCPower | OBW | PNOise | PSETtling | SPURious
|SEM }
```

Arguments `AM` deletes the AM view.

`FM` deletes the AM view.

`PM` deletes the AM view.

`CCDF` deletes the CCDF view.

`ACPower` deletes the Channel power and ACPR view.

`FSETtling` deletes the Frequency Settling view.

`MCPower` deletes the MCPR view.

OBW deletes the Occupied Bandwidth view.

PNOise deletes the Phase Noise view (Option 11 only).

PSETtling deletes the Phase Settling view.

SPURious deletes the Spurious view.

SEM deletes the Spectral Emissions Mask view.

If you attempt to delete a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running") will be returned.

Examples DISPLAY:GPRF:MEASVIEW:DELETEACPower deletes the Channel power and ACPR view.

DISPlay:GPRF:MEASview:NEW (No Query Form)

Displays a new measurement view in the RF measurements.

Conditions Measurement views: RF measurements

Group Display commands

Syntax

```
DISPlay:GPRF:MEASview:NEW { AM | FM | PM | CCDF | ACPower
| FSETtling MCPower | OBW | PNOise | PSETtling | SPURious
|SEM }
```

Arguments AM opens the AM view.

FM opens the AM view.

PM opens the AM view.

CCDF opens the CCDF view.

ACPower opens the Channel power and ACPR view.

FSETtling opens the Frequency Settling view.

MCPower opens the MCPR view.

OBW opens the Occupied Bandwidth view.

PNOise opens the Phase Noise view (Option 11 only).

PSETtling opens the Phase Settling view.

SPURious opens the Spurious view.

SEM opens the Spectral Emissions Mask view.

If you attempt to open a view that is currently displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Examples `DISPLAY:GPRF:MEASVIEW:NEWACPower` creates the Channel power and ACPR view.

DISPlay:GPRF:MEASview:SELect

Selects a measurement view in the RF measurements on the screen. The query command returns the currently selected view.

Selecting a measurement optimizes it. Other measurements may be optimized as a side effect. Refer to the DISPlay:WINDow:OPTimized:MEASurement? query.

Conditions Measurement views: RF measurements

Group Display commands

Syntax

```
DISPlay:GPRF:MEASview:SELect { AM | FM | PM | CCDF | ACPower
| FSETtling | MCPower | OBW | PNOise | PSETtling | SPURious
|SEM }
DISPlay:GPRF:MEASview:SELect?
```

Arguments

`AM` selects the AM view.

`FM` selects the AM view.

`PM` selects the AM view.

`CCDF` selects the CCDF view.

`ACPower` selects the Channel power and ACPR view.

`FSETtling` selects the Frequency Settling view.

`MCPower` selects the MCPR view.

`OBW` selects the Occupied Bandwidth view.

`PNOise` selects the Phase Noise view (Option 11 only).

`PSETtling` selects the Phase Settling view.

`SPURious` selects the Spurious view.

SEM selects the Spectral Emissions Mask view.

If you attempt to select a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running") will be returned.

Examples `DISPLAY:GPRF:MEASVIEW:SELECTACPower` selects the Channel power and ACPR view.

DISPlay:IQVTime:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions Measurement views: RF I&Q versus Time

Group Display commands

Syntax
```
DISPlay:IQVTime:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:IQVTime:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples `DISPLAY:IQVTIME:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the RF I&Q versus Time view.

DISPlay:IQVTime:X[:SCALe]

Sets or queries the horizontal scale (full-scale time) of the RF I&Q versus Time graph. Programming a specified scale sets DISPlay:IQVTime:X[:SCALe] AUTO:STATe OFF.

Conditions Measurement views: RF I&Q versus Time

Group Display commands

Syntax
```
DISPlay:IQVTime:X[:SCALe] <value>
DISPlay:IQVTime:X[:SCALe]?
```

Related Commands DISPlay:IQVTime:X[:SCALe]:AUTO:STATe, DISPlay:IQVTime:X[:SCALe]:OFFSet

Arguments `<value>::={ <NRf> | MAXimum | MINimum }` specifies the horizontal scale in full-scale time. `MAXimum` and `MINimum` represent the upper and lower limits of the setting range, respectively. Use the DISPlay:IQVTime:X[:SCALe]:MAXimum? and DISPlay:IQVTime:X[:SCALe]:MINimum? queries to get the upper and lower limit values of the setting range.

Examples `DISPLAY:IQVTIME:X:SCALE 100us` sets the horizontal scale to 100 µs.

DISPlay:IQVTime:X[:SCALe]:AUTO (No Query Form)

Sets the horizontal scale automatically to fit the waveform to the screen in the RF I&Q versus Time view. Executing this command sets DISPlay:IQVTime X[:SCALe]:AUTO:STATe ON.

Conditions Measurement views: RF I&Q versus Time

Group Display commands

Syntax `DISPlay:IQVTime:X[:SCALe]:AUTO`

Related Commands DISPlay:IQVTime:X[:SCALe], DISPlay:IQVTime:X[:SCALe]:AUTO:STATe

Arguments None

Examples `DISPLAY:IQVTIME:X:SCALE:AUTO` sets the horizontal scale automatically to fit the waveform to the screen.

DISPlay:IQVTime:X[:SCALe]:AUTO:STATe

Determines whether to set the horizontal scale automatically or manually.

Conditions Measurement views: RF I&Q versus Time

Group Display commands

Syntax

```
DISPlay:IQVTime:X[:SCALe]:AUTO:STATe { OFF | ON | 0 | 1 }
DISPlay:IQVTime:X[:SCALe]:AUTO:STATe?
```

Arguments

`OFF` or `0` specifies that the horizontal scale is set manually. To set it, use the DISPlay:IQVTime:X[:SCALe] and DISPlay:IQVTime:X[:SCALe]:OFFSet commands.

`ON` or `1` specifies that the horizontal scale is set automatically.

Examples

`DISPLAY:IQVTIME:X:SCALE:AUTO:STATE ON` specifies that the horizontal scale is set automatically.

DISPlay:IQVTime:X[:SCALe]:MAXimum? (Query Only)

Queries the upper limit of the horizontal scale setting range.

Conditions

Measurement views: RF I&Q versus Time

Group

Display commands

Syntax

```
DISPlay:IQVTime:X[:SCALe]:MAXimum?
```

Related Commands

DISPlay:IQVTime:X[:SCALe]

Arguments

None

Returns

`<NRf>` The upper limit of the horizontal scale setting range.

Examples

`DISPLAY:IQVTIME:X:SCALE:MAXIMUM?` might return `18.135E-3`, indicating that the upper limit of the horizontal scale setting range is 18.135 ms.

DISPlay:IQVTime:X[:SCALe]:MINimum? (Query Only)

Queries the lower limit of the horizontal scale setting range.

Conditions

Measurement views: RF I&Q versus Time

Group

Display commands

Syntax `DISPlay:IQVTime:X[:SCALe]:MINimum?`

Related Commands DISPlay:IQVTime:X[:SCALe]

Arguments None

Returns `<NRf>` The lower limit of the horizontal scale setting range.

Examples `DISPLAY:IQVTIME:X:SCALE:MINIMUM?` might return `10.0E-9`, indicating that the lower limit of the horizontal scale setting range is 10.0 ns.

DISPlay:IQVTime:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (left edge) of the RF I&Q versus Time graph. Programming a specified offset sets DISPlay:IQVTime:X[:SCALe] AUTO:STATe OFF.

Conditions Measurement views: RF I&Q versus Time

Group Display commands

Syntax
```
DISPlay:IQVTime:X[:SCALe]:OFFSet <value>
DISPlay:IQVTime:X[:SCALe]:OFFSet?
```

Related Commands DISPlay:IQVTime:X[:SCALe]:AUTO:STATe, DISPlay:IQVTime:X[:SCALe]

Arguments `<value>::={ <NRf> | MAXimum | MINimum }` specifies the horizontal offset. `MAXimum` and `MINimum` represent the upper and lower limits of the setting range, respectively.

Use the DISPlay:IQVTime:X[:SCALe]:OFFSet:MAXimum? and DISPlay:IQVTime:X[:SCALe]:OFFSet:MINimum? queries to get the upper and lower limit values of the setting range.

Examples `DISPLAY:IQVTIME:X:SCALE:OFFSET 800ns` sets the minimum horizontal value to 800 ns in the RF I&Q versus Time graph.

DISPlay:IQVTime:X[:SCALe]:OFFSet:MAXimum? (Query Only)

Queries the upper limit of the horizontal offset setting range.

Conditions Measurement views: RF I&Q versus Time

Group Display commands

Syntax `DISPlay:IQVTime:X[:SCALe]:OFFSet:MAXimum?`

Related Commands DISPlay:IQVTime:X[:SCALe]:OFFSet

Arguments None

Returns <NRf> The upper limit of the horizontal offset setting range.

Examples `DISPLAY:IQVTIME:X:SCALE:OFFSET:MAXIMUM?` might return `-1.812E-3`, indicating that the upper limit of the horizontal offset setting range is -1.812 ms.

DISPlay:IQVTime:X[:SCALe]:OFFSet:MINimum? (Query Only)

Queries the lower limit of the horizontal offset setting range.

Conditions Measurement views: RF I&Q versus Time

Group Display commands

Syntax `DISPlay:IQVTime:X[:SCALe]:OFFSet:MINimum?`

Related Commands DISPlay:IQVTime:X[:SCALe]:OFFSet

Arguments None

Returns <NRf> The lower limit of the horizontal offset setting range.

Examples `DISPLAY:IQVTIME:X:SCALE:OFFSET:MINIMUM?` might return `-16.28E-3`, indicating that the lower limit of the horizontal offset setting range is -16.28 ms.

DISPlay:IQVTime:Y[:SCALe]

Sets or queries the vertical range of the RF I&Q versus Time graph.

Conditions Measurement views: RF I&Q versus Time

Group Display commands

Syntax
```
DISPlay:IQVTime:Y[:SCALe] <value>
DISPlay:IQVTime:Y[:SCALe]?
```

Related Commands DISPlay:IQVTime:Y[:SCALe]:OFFSet

Arguments `<value> ::` `<NRf>` specifies the vertical range. Range: 1µ to 10 V.

Examples `DISPLAY:IQVTIME:Y:SCALE 1.5` sets the vertical range to 1.5 V in the RF I&Q versus Time graph.

DISPlay:IQVTime:Y[:SCALe]:AUTO (No Query Form)

Sets the vertical scale automatically to fit the waveform to the screen in the RF I&Q versus Time view.

Conditions Measurement views: RF I&Q versus Time

Group Display commands

Syntax `DISPlay:IQVTime:Y[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:IQVTIME:Y:SCALE:AUTO` sets the vertical scale automatically to fit the waveform to the screen.

DISPlay:IQVTime:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the center of the vertical axis) in the RF I&Q versus Time graph.

Conditions Measurement views: RF I&Q versus Time

Group Display commands

Syntax

```
DISPlay:IQVTime:Y[:SCALe]:OFFSet <value>
DISPlay:IQVTime:Y[:SCALe]:OFFSet?
```

Related Commands DISPlay:IQVTime:Y[:SCALe]

Arguments `<value> ::` `<NRf>` specifies the vertical offset. Range: -5 to +5 V.

Examples `DISPLAY:IQVTIME:Y:SCALE:OFFSET -82.75mV` sets the vertical offset to -82.75 mV in the RF I&Q versus Time graph.

DISPlay:IQVTime:Y[:SCALe]:RESCale (No Query Form)

Rescales the vertical axis automatically to fit the waveform to the screen.

Conditions Measurement views: RF I&Q versus Time

Group Display commands

Syntax

```
DISPlay:IQVTime:Y[:SCALe]:RESCale
```

Arguments None

Examples `DISPLAY:IQVTIME:Y:SCALE:RESCALE` rescales the vertical axis automatically to fit the waveform to the screen.

DISPlay:LTE:ACLR:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the LTE ACLR display. It will enable or disable the checkbox in the Prefs tab of the LTE ACLR settings control panel. This command will not add the marker itself. It will only decide if the enabled marker readout is shown or hidden.

Conditions Measurement view: LTE ACLR

Group Display commands

Syntax

```
DISPlay:LTE:ACLR:MARKer:SHOW:STATe {OFF | ON | 1 | 0}
DISPlay:LTE:ACLR:MARKer:SHOW:STATe?
```

Arguments OFF or 0 hides the readout for the selected marker in the LTE ACLR display.

ON or 1 shows the readout for the selected marker in the LTE ACLR display.

Returns OFF or 0 means the readout for the selected marker in the LTE ACLR display is hidden.

ON or 1 means the readout for the selected marker in the LTE ACLR display is showing.

Examples `DISPLAY:LTE:ACLR:MARKER:SHOW:STATE ON` shows the readout for the selected marker in the display if the marker is enabled.

DISPlay:LTE:ACLR:RESet:SCALe (No Query Form)

Resets the horizontal and vertical scale of the LTE ACLR display graph to the default values.

Conditions Measurement view: LTE ACLR

Group Display commands

Syntax

```
DISPlay:LTE:ACLR:RESet:SCALe
```

Arguments None

Examples `DISPLAY:LTE:ACLR:RESET:SCALE` resets the horizontal and vertical scale of the graph to the default values.

DISPlay:LTE:ACLR:SHOW:LIMit

Sets or queries the appearance of the limits in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Display commands

Syntax
```
DISPlay:LTE:ACLR:SHOW:LIMit { SHADed | LINE | OFF }
DISPlay:LTE:ACLR:SHOW:LIMit?
```

Arguments SHADed: denotes the limit area with shading.

LINE: delineates the limit area with a line.

OFF: turns off limits in the display.

Returns SHADED: means that the limit area is denoted with shading.

LINE: means that the limit area is delineated with a line.

OFF: means that the display of limits is off.

Examples `DISPLAY:LTE:ACLR:SHOW:LIMIT SHAD` sets the instrument to use shading to identify the limits of the graph in the display.

DISPlay:LTE:ACLR:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticules in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Display commands

Syntax
```
DISPlay:LTE:ACLR:WINDow:TRACe:GRATicule:GRID:STATe {OFF |
ON | 1 | 0}
DISPlay:LTE:ACLR:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments OFF or 0 hides the graticules in the graph.

ON or 1 shows the graticules in the graph.

Returns 0 indicates that the graticules in the graph is hidden.

1 indicates that the graticules in the graph is showing.

Examples `DISPLAY:LTE:ACLR:WINDOW:TRACE:GRATICULE:GRID:STATE OFF` hides the graticules in the graph.

DISPlay:LTE:ACLR:WINDow:TRACe:POWer:LEVEl:STATe

Determines whether to show or hide the power readout in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Display commands

Syntax

```
DISPlay:LTE:ACLR:WINDow:TRACe:POWer:LEVEl:STATe {OFF | ON |
1 | 0}
DISPlay:LTE:ACLR:WINDow:TRACe:POWer:LEVEl:STATe?
```

Arguments OFF or 0 hides the power readout in the display.

ON or 1 shows the power readout in the display.

Returns 0 indicates that the power readout in the display is hidden.

1 indicates that the power readout in the display is showing.

Examples `DISPLAY:LTE:ACLR:WINDOW:TRACE:POWER:LEVEL:STATE?` might return 1, indicating that the power readout in the display is showing.

DISPlay:LTE:ACLR:X:SCALe

Sets or queries the horizontal range of the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Display commands

Syntax
```
DISPlay:LTE:ACLR:X:SCALe <value>
DISPlay:LTE:ACLR:X:SCALe?
```

Arguments <value>::<NRf> specifies the horizontal range

Returns <NR3>

Examples `DISPLAY:LTE:ACLR:X:SCALE 10.000000000E+6` sets the horizontal range to 10 MHz.

DISPLay:LTE:ACLR:X:SCALe:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Display commands

Syntax
```
DISPLay:LTE:ACLR:X:SCALe:AUTO
```

Arguments None

Examples `DISPLAY:LTE:ACLR:X:SCALE:AUTO` rescales the horizontal axis to fit the display screen.

DISPlay:LTE:ACLR:X:SCALe:OFFSet

Sets or queries the center frequency of the trace in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Display commands

Syntax

```
DISPlay:LTE:ACLR:X:SCALe:OFFSet <value>
DISPlay:LTE:ACLR:X:SCALe:OFFSet?
```

Arguments <value>::=<NRf> specifies the center frequency.

Returns <NR3>

Examples `DISPLAY:LTE:ACLR:X:SCAEL:OFFSET 1.450000000E+9` sets the center frequency to 1.45 GHz.

DISPlay:LTE:ACLR:Y:SCALe:AUTO (No Query Form)

Rescales the vertical scale automatically to fit the waveform on the screen in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Display commands

Syntax

```
DISPlay:LTE:ACLR:Y:SCALe:AUTO
```

Arguments None

Examples `DISPLAY:LTE:ACLR:Y:SCALE:AUTO` rescales the vertical scale to fit the waveform on the screen.

DISPlay:LTE:ACLR:Y:SCALe:OFFSet

Sets or queries the vertical offset of the trace in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Display commands

Syntax

```
DISPlay:LTE:ACLR:Y:SCALe:OFFSet <value>
DISPlay:LTE:ACLR:Y:SCALe:OFFSet?
```

Arguments <value>::=<NRf> specifies the vertical offset.

Returns <NRf>

Examples `DISPLAY:LTE:ACLR:Y:SCALE:OFFSET` 40 sets the vertical position of the trace to 40 dBm.

DISPlay:LTE:ACLR:Y:SCALe

Sets or queries the vertical range of the graph in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Display commands

Syntax
```
DISPlay:LTE:ACLR:Y:SCALe <value>
DISPlay:LTE:ACLR:Y:SCALe?
```

Arguments <value>::=<NRf> specifies the vertical range.

Returns <NRf>

Examples `DISPLAY:LTE:ACLR:Y:SCALE` 100 sets the vertical range to 100 dB.

DISPlay:LTE:CHSPectrum:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the LTE Channel Spectrum display. It will enable or disable the checkbox in the Prefs tab of the LTE Channel Spectrum settings control panel. This command will not add the marker itself. It will only decide if the enabled marker readout is shown or hidden.

Conditions Measurement view: LTE Channel Spectrum

Group Display commands

Syntax

```
DISPlay:LTE:CHSPectrum:MARKer:SHOW:STATe {OFF | ON | 1 | 0}
DISPlay:LTE:CHSPectrum:MARKer:SHOW:STATe?
```

Arguments

OFF or 0 hides the readout for the selected marker in the LTE Channel Spectrum display.

ON or 1 shows the readout for the selected marker in the LTE Channel Spectrum display.

Returns

OFF or 0 means the readout for the selected marker in the LTE Channel Spectrum display is hidden.

ON or 1 means the readout for the selected marker in the LTE Channel Spectrum display is showing.

Examples

`DISPLAY:LTE:CHSPECTRUM:MARKER:SHOW:STATE ON` shows the readout for the selected marker in the display if the marker is enabled.

DISPlay:LTE:CHSPectrum:RESet:SCALe (No Query Form)

Resets the horizontal and vertical scale of the LTE Channel Spectrum display graph to the default values.

Conditions

Measurement view: LTE Channel Spectrum

Group

Display commands

Syntax

```
DISPlay:LTE:CHSPectrum:RESet:SCALe
```

Arguments

None

Examples

`DISPLAY:LTE:CHSPECTRUM:RESET:SCALE` resets the horizontal and vertical scale of the graph to the default values.

DISPlay:LTE:CHSPectrum:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule in the LTE Channel Spectrum display.

Conditions

Measurement view: LTE Channel Spectrum

Group Display commands

Syntax
```
DISPlay:LTE:CHSPectrum:WINDow:TRACe:GRATicule:GRID:STATe
{OFF | ON | 1 | 0}
DISPlay:LTE:CHSPectrum:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments OFF or 0 hides the graticule in the LTE Channel Spectrum display.

ON or 1 shows the graticule in the LTE Channel Spectrum display.

Returns OFF or 0 means the graticule in the LTE Channel Spectrum display is hidden.

ON or 1 means the graticule in the LTE Channel Spectrum display is showing.

Examples `DISPLAY:LTE:CHSPECTRUM:WINDOW:TRACE:GRATICULE:GRID:STATE OFF` hides the graticules in the LTE Channel Spectrum display.

DISPlay:LTE:CHSPectrum:X:SCALe

Sets or queries the horizontal range of the LTE Channel Spectrum display.

Conditions Measurement view: LTE Channel Spectrum

Group Display commands

Syntax
```
DISPlay:LTE:CHSPectrum:X:SCALe <value>
DISPlay:LTE:CHSPectrum:X:SCALe?
```

Arguments <value>::<NRf> specifies the horizontal range

Returns <NR3>

Examples `DISPLAY:LTE:CHSPECTRUM:X:SCALE 10.000000000E+6` sets the horizontal range to 10 MHz.

DISPLay:LTE:CHSpectrum:X:SCALe:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the LTE Channel Spectrum display.

Conditions Measurement view: LTE Channel Spectrum

Group Display commands

Syntax `DISPLay:LTE:CHSpectrum:X:SCALe:AUTO`

Arguments None

Examples `DISPLAY:LTE:CHSPECTRUM:X:SCALE:AUTO` rescales the horizontal axis to fit the display screen.

DISPlay:LTE:CHSPectrum:X:SCALe:OFFSet

Sets or queries the center frequency of the trace in the LTE Channel Spectrum display.

Conditions Measurement view: LTE Channel Spectrum

Group Display commands

Syntax
```
DISPlay:LTE:CHSPectrum:X:SCALe:OFFSet <value>
DISPlay:LTE:CHSPectrum:X:SCALe:OFFSet?
```

Arguments <value>::=<NRf> specifies the center frequency.

Returns <NR3>

Examples `DISPLAY:LTE:ACLR:X:SCAEL:OFFSET 1.450000000E+9` sets the center frequency to 1.45 GHz.

DISPlay:LTE:CHSPectrum:Y:SCALe

Sets or queries the vertical range of the graph in the LTE Channel Spectrum display.

Conditions Measurement view: LTE Channel Spectrum

Group Display commands

Syntax
```
DISPlay:LTE:CHSPectrum:Y:SCALe <value>
DISPlay:LTE:CHSPectrum:Y:SCALe?
```

Arguments <value>::=<NRf> specifies the vertical range.

Returns <NRf>

Examples `DISPLAY:LTE:CHSPECTRUM:Y:SCALE 100` sets the vertical range to 100 dB.

DISPlay:LTE:CHSPectrum:Y:SCALe:AUTO (No Query Form)

Rescales the vertical scale automatically to fit the waveform on the screen in the LTE Channel Spectrum display.

Conditions Measurement view: LTE Channel Spectrum

Group Display commands

Syntax
```
DISPlay:LTE:CHSPectrum:Y:SCALe:AUTO
DISPlay:LTE:CHSPectrum:Y:SCALe:AUTO?
```

Arguments None

Examples `DISPLAY:LTE:CHSPECTRUM:Y:SCALE:AUTO` rescales the vertical scale to fit the waveform on the screen.

DISPlay:LTE:CHSPectrum:Y:SCALe:OFFSet

Sets or queries the vertical offset of the trace in the LTE Channel Spectrum display.

Conditions Measurement view: LTE Channel Spectrum

Group Display commands

Syntax

```
DISPlay:LTE:CHSPectrum:Y:SCALe:OFFSet <value>
DISPlay:LTE:CHSPectrum:Y:SCALe:OFFSet?
```

Arguments <value>::=<NRf> specifies the vertical offset.

Returns <NRf>

Examples `DISPLAY:LTE:CHSPECTRUM:Y:SCALE:OFFSET` 40 sets the vertical position of the trace to 40 dBm.

DISPlay:LTE:CONSte:SCALe

Sets or queries the scale (zoom) in the LTE Constellation display.

Conditions Measurement view: LTE Constellation

Group Display commands

Syntax

```
DISPlay:LTE:CONSte:SCALe <value>
DISPlay:LTE:CONSte:SCALe?
```

Arguments <value>::<NRf> specifies the scale (zoom)

Returns <NRf>

Examples `DISPLAY:LTE:CONSTE:SCALE` 5 zooms the display out by 5.

DISPlay:LTE:CONSte:X:OFFSet

Sets or queries the horizontal offset in the LTE Constellation display.

Conditions Measurement view: LTE Constellation

Group Display commands

Syntax

```
DISPlay:LTE:CONSte:X:OFFSet <value>
DISPlay:LTE:CONSte:X:OFFSet?
```

Arguments <value>::<NRf> specifies the horizontal offset range.

Returns <NRf>

Examples `DISPLAY:LTE:CONSTE:X:OFFSET 2.7` sets the horizontal offset to 2.7 units to the left.

DISPlay:LTE:CONSte:Y:OFFSet

Sets or queries the vertical offset in the LTE Constellation display.

Conditions Measurement view: LTE Constellation

Group Display commands

Syntax

```
DISPlay:LTE:CONSte:Y:OFFSet <value>
DISPlay:LTE:CONSte:Y:OFFSet?
```

Arguments <value>::<NRf> specifies the vertical offset range. A positive value moves the offset up. A negative value moves the offset down and the display moves in the other direction.

Returns <NRf>

Examples `DISPLAY:LTE:CONSTE:Y:OFFSET 2` sets the vertical offset to 2 units.

DISPlay:LTE:MEASview:DELete (No Query Form)

Closes the selected LTE measurement display in the application.

Conditions Measurement view: Any LTE display

Group Display commands

Syntax `DISPlay:LTE:MEASview:DELete { CONS | ACLR | CHSP | PVT }`

Arguments CONS closes the LTE Constellation display.

ACLR closes the LTE ACLR display.

CHSP closes the LTE Channel Spectrum display.

PVT closes the LTE Power vs Time display.

Examples `DISPLAY:LTE:MEASVIEW:DELETE ACLR` closes the LTE ACLR display.

DISPlay:LTE:MEASview:NEW (No Query Form)

Opens a new LTE measurement display.

Conditions Measurement view: Any LTE display

Group Display commands

Syntax `DISPlay:LTE:MEASview:NEW { CONS | ACLR | CHSP | PVT }`

Arguments CONS opens the LTE Constellation display.

ACLR opens the LTE ACLR display.

CHSP opens the LTE Channel Spectrum display.

PVT opens the LTE Power vs Time display.

Examples `DISPLAY:LTE:MEASVIEW:NEW ACLR` opens a new LTE ACLR display.

DISPlay:LTE:MEASview:SELect

Selects an LTE measurement display in the application. The query returns the currently selected display.

Conditions Measurement view: Any LTE display

Group Display commands

Syntax

```
DISPlay:LTE:MEASview:SELect { CONS | ACLR | CHSP | PVT }
DISPlay:LTE:MEASview:SELect?
```

Arguments

CONS selects the LTE Constellation display.

ACLR selects the LTE ACLR display.

CHSP selects the LTE Channel Spectrum display.

PVT selects the LTE Power vs Time display.

Returns

CONS means the LTE Constellation display is selected.

ACLR means the LTE ACLR display is selected.

CHSP means the LTE Channel Spectrum display is selected.

PVT means the LTE Power vs Time display is selected.

Examples

`DISPLAY:LTE:MEASVIEW:SELECT ACLR` selects the LTE ACLR display.

DISPlay:LTE:PVTime:MARKer:SHOW:STATe

Shows or hides the marker readout on the LTE Power vs. Time display.

Conditions

Measurement view: LTE Power vs Time

Group

Display commands

Syntax

```
DISPlay:LTE:PVTime:MARKer:SHOW:STATe {OFF | ON | 0 | 1 }
DISPlay:LTE:PVTime:MARKer:SHOW:STATe?
```

Arguments

OFF or 0 hides the marker readout in the LTE Power vs Time display.

ON or 1 shows the marker readout in the LTE Power vs Time display.

Returns

OFF or 0 means the marker readout in the LTE Power vs Time display is hidden.

ON or 1 means the marker readout in the LTE Power vs Time display is showing.

Examples

`DISPLAY:LTE:PVTIME:MARKER:SHOW:STATE ON` turns on the marker readout.

DISPlay:LTE:PVTime:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule in the LTE Power vs Time display.

Conditions Measurement view: LTE Power vs Time

Group Display commands

Syntax

```
DISPlay:LTE:PVTime:WINDow:TRACe:GRATicule:GRID:STATe {OFF | ON | 0 | 1 }
DISPlay:LTE:PVTime:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments OFF or 0 hides the graticule in the LTE Power vs Time display.

ON or 1 shows the graticule in the LTE Power vs Time display.

Returns OFF or 0 means the graticule in the LTE Power vs Time display is hidden.

ON or 1 means the graticule in the LTE Power vs Time display is showing.

Examples `DISPLAY:LTE:PVTIME:WINDOW:TRACE:GRATICULE:GRID:STATE OFF` hides the graticules in the LTE Power vs Time display.

DISPlay:LTE:PVTime:X:SCALe

Sets or queries the vertical range of the LTE Power vs Time graph.

Conditions Measurement view: LTE Power vs Time

Group Display commands

Syntax

```
DISPlay:LTE:PVTime:X:SCALe <NRf>
DISPlay:LTE:PVTime:X:SCALe?
```

Arguments <NRf> specifies the vertical range.

Returns <NRf>

Examples `DISPLAY:LTE:PVTIME:X:SCALE 10` sets the horizontal range to 10 dB.

DISPLay:LTE:PVTime:X:SCALe:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the LTE Power vs Time display.

Conditions Measurement view: LTE Power vs Time

Group Display commands

Syntax `DISPLay:LTE:PVTime:X:SCALe:AUTO`

Arguments None

Examples `DISPLAY:LTE:PVTIME:X:SCALE:AUTO` rescales the horizontal axis to fit the display screen.

DISPlay:LTE:PVTime:X:SCALe:RESet (No Query Form)

Resets the horizontal scale to the default value for the time measurements for the LTE Power vs Time display.

Conditions Measurement view: LTE Power vs Time

Group Display commands

Syntax `DISPlay:LTE:PVTime:X:SCALe:RESet`

Arguments None

Examples `DISPLAY:LTE:PVTIME:X:SCALE:RESET` resets the horizontal scale to the default value.

DISPlay:LTE:PVTime:Y:SCALe

Sets or queries the horizontal range of the LTE Power vs Time graph.

Conditions Measurement view: LTE Power vs Time

Group Display commands

Syntax
```
DISPlay:LTE:PVTime:Y:SCALe <value>
DISPlay:LTE:PVTime:Y:SCALe?
```

Arguments <value>::<NRf> specifies the horizontal range.

Returns <NRf>

Examples `DISPLAY:LTE:PVTIME:Y:SCALE 100` sets the horizontal range to 100 ms.

DISPlay:LTE:PVTime:Y:SCALe:AUTO (No Query Form)

Rescales the vertical scale automatically to fit the waveform on the screen in the LTE Power vs Time display.

Conditions Measurement view: LTE Power vs Time

Group Display commands

Syntax `DISPlay:LTE:PVTime:Y:SCALe:AUTO`

Arguments None

Examples `DISPLAY:LTE:PVTIME:Y:SCALE:AUTO` rescales the vertical scale to fit the waveform on the screen.

DISPlay:LTE:PVTime:Y:SCALe:OFFSet

Sets or queries the vertical offset (top edge of the vertical axis) of the trace in the LTE Power vs Time display.

Conditions Measurement view: LTE Power vs Time

Group Display commands

Syntax
```
DISPlay:LTE:PVTime:Y:SCALe:OFFSet <NRf>
DISPlay:LTE:PVTime:Y:SCALe:OFFSet?
```

Arguments <NRf> specifies the vertical offset.

Returns <NRf>

Examples `DISPLAY:LTE:PVTIME:Y:SCALE:OFFSET` –10 sets the vertical position of the trace to –10 dBm.

DISPlay:LTE:PVTime:Y:SCALe:PDIVision

Sets or queries the value of the vertical scale for the LTE Power vs. Time display in dB/division. This is only a visual control for panning the graph.

Conditions Measurement view: LTE Power vs Time

Group Display commands

Syntax
```
DISPlay:LTE:PVTime:Y:SCALe:PDIVision <NRf>
DISPlay:LTE:PVTime:Y:SCALe:PDIVision?
```

Arguments <NRf> specifies the vertical scale in dB/div.

Returns <NRf>

Examples `DISPLAY:LTE:PVTIME:Y:SCALE:PDIVISION 20` sets the vertical scale to 20.0 dB/div.

DISPlay:LTE:PVTime:Y:SCALe:RESet (No Query Form)

Resets the vertical scale and position (offset) to the default value for the time measurements for the LTE Power vs Time display.

Conditions Measurement view: LTE Power vs Time

Group Display commands

Syntax `DISPlay:LTE:PVTime:Y:SCALe:RESet`

Arguments None

Examples `DISPLAY:LTE:PVTIME:Y:SCALE:RESET` resets the vertical scale and position (offset) to the default value.

DISPlay:MCPower:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the MCPR view.

Conditions Measurement views: MCPR

Group Display commands

Syntax
```
DISPlay:MCPower:MARKer:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:MCPower:MARKer:SHOW:STATe?
```

Arguments `OFF` or `0` hides the readout for the selected marker in the graph.

`ON` or `1` shows the readout for the selected marker in the graph.

Examples `DISPLAY:MCPOWER:MARKER:SHOW:STATE ON` shows the readout for the selected marker in the graph.

DISPlay:MCPower:PLEVel:SHOW:STATe

Determines whether to show or hide the power levels in the MCPR view.

Conditions Measurement views: MCPR

Group Display commands

Syntax

```
DISPlay:MCPower:PLEVel:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:MCPower:PLEVel:SHOW:STATe?
```

Arguments

`OFF` or `0` hides the power levels in the graph.

`ON` or `1` shows the power levels in the graph.

Examples

`DISPLAY:MCPOWER:PLEVEL:SHOW:STATE ON` shows the power levels in the graph.

DISPlay:MCPower:RESet:SCALe (No Query Form)

Resets the horizontal and vertical scale to the default values described below in the MCPR view.

Vertical offset = Reference level,
Vertical scale = 100 dB,
Horizontal offset = Center frequency, and
Horizontal scale = Default span

Conditions Measurement views: MCPR

Group Display commands

Syntax

```
DISPlay:MCPower:RESet:SCALe
```

Arguments None

Examples

`DISPLAY:MCPOWER:RESET:SCALE` resets the horizontal and vertical scale to the default values.

DISPlay:MCPower:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions Measurement views: MCPR

Group Display commands

Syntax

```
DISPlay:MCPower:WINDow:TRACe:GRATicule:GRID:STATe { OFF |
ON | 0 | 1 }
DISPlay:MCPower:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples

`DISPLAY:MCPOWER:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the screen.

DISPlay:MCPower:X[:SCALe]

Sets or queries the horizontal range of the MCPR graph.

Conditions

Measurement views: MCPR

Group

Display commands

Syntax

```
DISPlay:MCPower:X[:SCALe] <value>
DISPlay:MCPower:X[:SCALe]?
```

Related Commands

DISPlay:MCPower:X[:SCALe]:OFFSet

Arguments

`<value> ::  <NRf>` specifies the horizontal range.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

`DISPLAY:MCPOWER:X:SCALE 10MHz` sets the horizontal range to 10 MHz.

DISPlay:MCPower:X[:SCALe]:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the MCPR view.

Conditions Measurement views: MCPR

Group Display commands

Syntax `DISPlay:MCPower:X[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:MCPOWER:X:SCALE:AUTO` rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:MCPower:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (left edge) of the MCPR graph.

Conditions Measurement views: MCPR

Group Display commands

Syntax
```
DISPlay:MCPower:X[:SCALe]:OFFSet <value>
DISPlay:MCPower:X[:SCALe]:OFFSet?
```

Related Commands DISPlay:MCPower:X[:SCALe]

Arguments `<value> ::= <NRf>` specifies the minimum horizontal value.
Range: [(center frequency) - (X scale) × 0.9] to [(center frequency) + (X scale) × 0.9]

Examples `DISPLAY:MCPOWER:X:SCALE:OFFSET 1.45GHz` sets the minimum horizontal value to 1.45 GHz in the MCPR graph.

DISPlay:MCPower:Y[:SCALe]

Sets or queries the vertical range of the MCPR graph.

Conditions Measurement views: MCPR

Group Display commands

Syntax
```
DISPlay:MCPower:Y[:SCALe] <value>
DISPlay:MCPower:Y[:SCALe]?
```

Related Commands DISPlay:MCPower:Y[:SCALe]:OFFSet

Arguments `<value> ::` `<NRf>` specifies the vertical range. Range: 0.1 to 200 dB.

Examples `DISPLAY:MCPOWER:Y:SCALE 100` sets the vertical range to 100 dB in the MCPR graph.

DISPlay:MCPower:Y[:SCALe]:AUTO (No Query Form)

Rescales the vertical axis automatically to fit the waveform to the screen in the MCPR view.

Conditions Measurement views: MCPR

Group Display commands

Syntax `DISPlay:MCPower:Y[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:MCPOWER:Y:SCALE:AUTO` rescales the vertical scale automatically to fit the waveform to the screen.

DISPlay:MCPower:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the top edge of the vertical axis) in the MCPR graph.

Conditions Measurement views: MCPR

Group Display commands

Syntax

```
DISPlay:MCPower:Y[:SCALe]:OFFSet <value>
DISPlay:MCPower:Y[:SCALe]:OFFSet?
```

Related Commands DISPlay:MCPower:Y[:SCALe]

Arguments `<value> ::` `<NRf>` specifies the vertical offset. Range: -170 to +50 dBm.

Examples `DISPLAY:MCPOWER:Y:SCALE:OFFSET -12.5` sets the vertical offset to -12.5 dBm in the MCPR graph.

DISPlay:MERRor:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries the graticule grid view state for the Magnitude Error versus Time view.

Conditions Measurement views: Magnitude error versus Time

Group Display commands

Syntax

```
DISPlay:MERRor:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON
| 0 | 1 }
DISPlay:MERRor:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples `DISPLAY:MERROR:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the Magnitude Error versus Time view.

DISPlay:MERRor:Y[:SCALe]

Sets or queries the vertical range of the Magnitude Error versus Time graph.

Conditions Measurement views: Magnitude Error versus Time

Group Display commands

Syntax

```
DISPlay:MERRor:Y[:SCALe] <value>
DISPlay:MERRor:Y[:SCALe]?
```

Related Commands DISPlay:MERRor:Y[:SCALe]:OFFSet

Arguments <value> :: <NRf> specifies the vertical range. Range: 1 to 100%.

Examples DISPLAY:MERROR:Y:SCALE 50 sets the vertical range to 50% in the Magnitude error versus Time graph.

DISPlay:MERRor:Y[:SCALe]:AUTO (No Query Form)

Sets the vertical scale automatically to fit the waveform to the screen in the Magnitude error versus Time view.

Conditions Measurement views: Magnitude error versus Time

Group Display commands

Syntax

```
DISPlay:MERRor:Y[:SCALe]:AUTO
```

Arguments None

Examples DISPLAY:MERROR:Y:SCALE:AUTO sets the vertical scale automatically to fit the waveform to the screen.

DISPlay:MERRor:Y[:SCALe]:OFFSet

Sets or queries the minimum vertical value (bottom edge) of the Magnitude error versus Time graph.

Conditions Measurement views: Magnitude error versus Time

Group Display commands

Syntax

```
DISPlay:MERRor:Y[:SCALe]:OFFSet <value>
DISPlay:MERRor:Y[:SCALe]:OFFSet?
```

Related Commands DISPlay:MERRor:Y[:SCALe]

Arguments `<value> ::= <NRf>` specifies the minimum vertical value. Range: -100 to +100%.

Examples `DISPLAY:MERROR:Y:SCALE:OFFSET -9.5` sets the minimum vertical value to -9.5% in the Magnitude error versus Time graph.

DISPlay:NOISe:FIGure[:SCALe]:AUTO (No Query Form)

Automatically scales the Noise Figure graph.

Conditions Measurement view: Noise Figure

Group Display commands

Syntax `DISPlay:NOISe:FIGure[:SCALe]:AUTO`

Arguments None

Examples `DISP:NOIS:FIG:SCAL:AUTO` will automatically scale the Noise Figure graph.

DISPlay:NOISe:FIGure:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries the graticule state in the Noise Figure display to on (showing) or off (hidden).

Conditions Measurement view: Noise Figure

Group Display commands

Syntax
```
DISPlay:NOISe:FIGure:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:FIGure:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule in the Noise Figure display.

`ON` or `1` shows the graticule in the Noise Figure display.

Returns 0 means the graticule in the Noise Figure display is hidden (off).

1 means the graticule in the Noise Figure display is showing (on).

Examples DISPLAY:NOISE:FIGURE:WINDOW:TRACE:GRATICULE:GRID:STATE ON will turn on the graticule in the Noise Figure display.

DISPLAY:NOISE:FIGURE:WINDOW:TRACE:GRATICULE:GRID:STATE? 1 indicates that the graticule in the Noise Figure display is showing (on).

DISPlay:NOISe:FIGure:WINDow:TRACe:LEGend:STATe

Sets or queries the trace legend state in the Noise Figure display to on (showing) or off (hidden).

Conditions Measurement view: Noise Figure

Group Display commands

Syntax
```
DISPlay:NOISe:FIGure:WINDow:TRACe:LEGend:STATe { OFF | ON
| 0 | 1 }
DISPlay:NOISe:FIGure:WINDow:TRACe:LEGend:STATe?
```

Arguments OFF or 0 hides the trace legend in the Noise Figure display.

ON or 1 shows the trace legend in the Noise Figure display.

Returns 0 means the trace legend in the Noise Figure display is hidden (off).

1 means the trace legend in the Noise Figure display is showing (on).

Examples DISPLAY:NOISE:FIGURE:WINDOW:TRACE:LEGEND:STATE ON will turn on the trace legend in the Noise Figure display.

DISPLAY:NOISE:FIGURE:WINDOW:TRACE:LEGEND:STATE? 1 indicates that the trace legend in the Noise Figure display is showing (on).

DISPlay:NOISe:FIGure:WINDow:TRACe:MEASpoints:STATe

Sets or queries the measurement points state in the Noise Figure display to on (showing) or off (hidden).

Conditions Measurement view: Noise Figure

Group Display commands

Syntax
```
DISPlay:NOISe:FIGure:WINDow:TRACe:MEASpoints:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:FIGure:WINDow:TRACe:MEASpoints:STATe?
```

Arguments `OFF` or `0` hides the measurement points in the Noise Figure display.

`ON` or `1` shows the measurement points in the Noise Figure display.

Returns `0` means the measurement points in the Noise Figure display is hidden (off).

`1` means the measurement points in the Noise Figure display is showing (on).

Examples `DISPLAY:NOISE:FIGURE:WINDOW:TRACE:MEASPOINTS:STATE ON` will turn on the measurement points in the Noise Figure display.

`DISPLAY:NOISE:FIGURE:WINDOW:TRACE:MEASPOINTS:STATE?` might return 1, indicating that the measurement points in the Noise Figure display are showing (on).

DISPlay:NOISe:FIGure:X:AUTO (No Query Form)

Automatically sets the horizontal scale of the Noise Figure graph.

Conditions Measurement view: Noise Figure

Group Display commands

Syntax
```
DISPlay:NOISe:FIGure:X:AUTO
```

Arguments None

DISPlay:NOISe:FIGure:X:OFFSet

Sets or queries the center frequency (offset) in the Noise Figure display.

Conditions Measurement view: Noise Figure

Group Display commands

Syntax
```
DISPlay:NOISe:FIGure:X:OFFSet <NRf>
DISPlay:NOISe:FIGure:X:OFFSet?
```

Arguments <NRf> is the center frequency (offset) in the Noise Figure display.

Range: -1.9900 GHz to +5.0000 GHz

Returns <NR3> is a floating point value with an exponent. This is the center frequency (offset) value.

Examples `DISPLAY:NOISE:FIGURE:X:OFFSET?` might return `2.0000000009E+9`, indicating that the center frequency (offset) value is 2.0000 GHz.

DISPlay:NOISe:FIGure:X[:SCALe]

Sets or queries the horizontal scale of the Noise Figure display.

Conditions Measurement view: Noise Figure

Group Display commands

Syntax
```
DISPlay:NOISe:FIGure:X[:SCALe] <NRf>
DISPlay:NOISe:FIGure:X[:SCALe]
```

Arguments <NRf> is the horizontal scale value of the display.

Range: 100 Hz to 6.2 GHz

Returns <NR3> = a floating point value with an exponent. This is the horizontal scale value.

Examples `DISPLAY:NOISE:FIGURE:X:SCALE?` might return `2.9900000000E+9`, indicating that the horizontal scale value is 3.0 GHz.

DISPlay:NOISe:FIGure:Y:AUTO (No Query Form)

Automatically sets the vertical scale of the Noise Figure graph.

Conditions Measurement view: Noise Figure

Group Display commands

Syntax `DISPlay:NOISe:FIGure:Y:AUTO`

Arguments None

DISPlay:NOISe:FIGure:Y:AUTO:STATe

Sets the state of the automatic vertical scale and position of the graph to on or off. The query form of the command returns the current state.

Conditions Measurement view: Noise Figure

Group Display commands

Syntax
```
DISPlay:NOISe:FIGure:Y:AUTO:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:FIGure:Y:AUTO:STATe?
```

Arguments `OFF` or `0` sets the analyzer to manual setting of the vertical scale and position of the graph.

`ON` or `1` sets the analyzer to automatically set the vertical scale and position of the graph.

Returns `OFF` or `0` means the analyzer is in manual mode for setting of the vertical scale and position of the graph.

`ON` or `1` means the analyzer is in automatic mode for setting of the vertical scale and position of the graph.

Examples `DISPLAY:NOISE:FIGURE:Y:AUTO:STATE ON` sets the analyzer to automatically set the vertical scale and position of the graph.

DISPlay:NOISe:FIGure:Y:PDIVision

Sets or queries the distance between graticule lines on the display in dB.

Conditions Measurement view: Noise Figure

Group Display commands

Syntax

```
DISPlay:NOISe:FIGure:Y:PDIVision <NRf>
DISPlay:NOISe:FIGure:Y:PDIVision?
```

Arguments <NRf> is the distance between graticule lines on the display.

Range: 100 mdB to 5.00 dB

Returns <NR2> = a floating point value without an exponent, which is the distance between graticule lines on the display in dB.

Examples `DISPLAY:NOISE:FIGURE:Y:PDIVISION?` might return 1.5000000000, indicating that the distance between graticule lines on the display is 1.50 dB.

DISPlay:NOISe:FIGure:Y:POSition

Sets or queries the vertical position of the Noise Figure display.

Conditions Measurement view: Noise Figure

Group Display commands

Syntax

```
DISPlay:NOISe:FIGure:Y:POSition <NRf>
DISPlay:NOISe:FIGure:Y:POSition?
```

Arguments <NRf> is the vertical position of the Noise Figure display.

Range: 5.12 to 26.5

Returns <NR2> = a floating point value without an exponent, which is the vertical position value.

The vertical position value

Examples `DISPLAY:NOISE:FIGURE:Y:POSITION?` might return 26.5000000000, indicating that the vertical position is 26.5.

DISPlay:NOISe:FIGure:Y:SCALe

Sets or queries the vertical scale of the Noise Figure display.

Conditions Measurement view: Noise Figure

Group Display commands

Syntax
```
DISPlay:NOISe:FIGure:Y:SCALe <NRf>
DISPlay:NOISe:FIGure:Y:SCALe?
```

Arguments <NRf> is the vertical scale of the Noise Figure display

Range: 10.2 to 31.6

Returns <NR2> = a floating point value without an exponent, which is the vertical scale value.

Examples `DISPLAY:NOISE:FIGURE:Y:SCALE?` might return 10.0000000000, indicating that the vertical scale is 10.0 dB.

DISPlay:NOISe:FIGure:Y:SCALe:LINear:STATe

Sets or queries whether the linear units are shown (on) or not shown (off) on the display.

Conditions Measurement view: Noise Figure

Group Display commands

Syntax
```
DISPlay:NOISe:FIGure:Y:SCALe:LINear:STATe { OFF | ON | 0 |
1 }
DISPlay:NOISe:FIGure:Y:SCALe:LINear:STATe?
```

Arguments OFF or 0 sets the analyzer to show the linear units on the display.

ON or 1 sets the analyzer to not show the linear units on the display.

Returns OFF or 0 indicates that the analyzer is set to show the linear units on the display.

ON or 1 indicates that the analyzer is set to not show the linear units on the display.

Examples DISPLAY:NOISE:FIGURE:Y:SCALE:LINEAR:STATE? might return 1, indicating that linear units are not showing on the display.

DISPlay:NOISe:GAIN[:SCALe]:AUTO (No Query Form)

Automatically scales the Gain graph.

Conditions Measurement view: Gain

Group Display commands

Syntax
```
DISPlay:NOISe:GAIN[:SCALe]:AUTO
```

Arguments None

DISPlay:NOISe:GAIN:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries the graticule state in the Gain display to on (showing) or off (hidden).

Conditions Measurement view: Gain

Group Display commands

Syntax
```
DISPlay:NOISe:GAIN:WINDow:TRACe:GRATicule:GRID:STATe
{OFF|ON|0|1}
DISPlay:NOISe:GAIN:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments OFF or 0 hides the graticule in the Gain display.

ON or 1 shows the graticule in the Gain display.

Returns 0 means the graticule in the Gain display is hidden (off).

1 means the graticule in the Gain display is showing (on).

Examples DISPLAY:NOISE:GAIN:WINDOW:TRACE:GRATICULE:GRID:STATE ON will turn on the graticule in the Gain display.

DISPLAY:NOISE:GAIN:WINDOW:TRACE:GRATICULE:GRID:STATE? might return 1, indicating that the graticule in the Gain display is showing (on).

DISPlay:NOISe:GAIN:WINDow:TRACe:LEGend:STATe

Sets or queries the trace legend state in the Gain display to on (showing) or off (hidden).

Conditions Measurement view: Gain

Group Display commands

Syntax
```
DISPlay:NOISe:GAIN:WINDow:TRACe:LEGend:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:GAIN:WINDow:TRACe:LEGend:STATe?
```

Arguments OFF or 0 hides the trace legend in the Gain display.

ON or 1 shows the trace legend in the Gain display.

Returns 0 means the trace legend in the Gain display is hidden (off).

1 means the trace legend in the Gain display is showing (on).

Examples DISPLAY:NOISE:GAIN:WINDOW:TRACE:LEGEND:STATE ON will turn on the trace legend in the Gain display.

DISPLAY:NOISE:GAIN:WINDOW:TRACE:LEGEND:STATE? 1 indicates that the trace legend in the Gain display is showing (on).

DISPlay:NOISe:GAIN:WINDow:TRACe:MEASPoints:STATe

Sets or queries the measurement points state in the Gain display to on (showing) or off (hidden).

Conditions Measurement view: Gain

Group Display commands

Syntax
```
DISPlay:NOISe:GAIN:WINDow:TRACe:MEASPoints:STATe
{OFF|ON|0|1}
DISPlay:NOISe:GAIN:WINDow:TRACe:MEASPoints:STATe?
```

Arguments `OFF` or `0` hides the measurement points in the Gain display.

`ON` or `1` shows the measurement points in the Gain display.

Returns `0` means the measurement points in the Gain display is hidden (off).

`1` means the measurement points in the Gain display is showing (on).

Examples `DISPLAY:NOISE:GAIN:WINDOW:TRACE:MEASPOINTS:STATE ON` will turn on the measurement points in the Gain display.

`DISPLAY:NOISE:GAIN:WINDOW:TRACE:MEASPOINTS:STATE?` might return 1, indicating that the measurement points in the Gain display are showing (on).

DISPlay:NOISe:GAIN:X:AUTO (No Query Form)

Automatically sets the horizontal scale of the Gain graph.

Conditions Measurement view: Gain

Group Display commands

Syntax `DISPlay:NOISe:GAIN:X:AUTO`

Arguments None

DISPlay:NOISe:GAIN:X:OFFSet

Sets or queries the center frequency (offset) in the Gain display.

Conditions Measurement view: Gain

Group Display commands

Syntax
```
DISPlay:NOISe:GAIN:X:OFFSet <NRf>
DISPlay:NOISe:GAIN:X:OFFSet?
```

Arguments <NRf> is the center frequency (offset) in the Gain display.

Range: -1.9900 GHz to +5.0000 GHz

Returns <NR3> = a floating point value with an exponent, which is the center frequency (offset) value.

Examples `DISPLAY:NOISE:GAIN:X:OFFSET?` might return `2.0000000009E+9`, indicating that the center frequency (offset) value is 2.0000 GHz.

DISPlay:NOISe:GAIN:X:SCALe

Sets or queries the horizontal scale of the Gain display.

Conditions Measurement view: Gain

Group Display commands

Syntax
```
DISPlay:NOISe:GAIN:X:SCALe <NRf>
DISPlay:NOISe:GAIN:X:SCALe
```

Related Commands DISPlay:NOISe:GAIN:Y:SCALe

Arguments <NRf> is the horizontal scale of the Gain display.

Range: 100 Hz to 6.2 GHz

Returns <NR3> = a floating point value with an exponent, which is the horizontal scale value.

Examples `DISP:NOIS:GAIN:X:SCAL?` might return `2.9900000000E+9`, indicating that the horizontal scale value is 3.0 GHz.

DISPlay:NOISe:GAIN:Y:AUTO (No Query Form)

Automatically sets the vertical scale of the Gain graph.

Conditions Measurement view: Gain

Group Display commands

Syntax
```
DISPlay:NOISe:GAIN:Y:AUTO
```

Related Commands DISPlay:NOISe:GAIN:Y:AUTO:STATe

DISPlay:NOISe:GAIN:X:AUTO

Arguments None

DISPlay:NOISe:GAIN:Y:AUTO:STATe

Sets or queries the state of the automatic vertical scale and position of the Gain graph to on or off.

Conditions Measurement view: Gain

Group Display commands

Syntax
```
DISPlay:NOISe:GAIN:Y:AUTO:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:GAIN:Y:AUTO:STATe?
```

Related Commands DISPlay:NOISe:GAIN:Y:AUTO

Arguments `OFF` or `0` sets the analyzer to manual setting of the vertical scale and position of the graph.

`ON` or `1` sets the analyzer to automatically set the vertical scale and position of the graph.

Returns `OFF` or `0` means the analyzer is in manual mode for setting of the vertical scale and position of the graph.

`ON` or `1` means the analyzer is in automatic mode for setting of the vertical scale and position of the graph.

Examples `DISPLAY:NOISE:GAIN:Y:AUTO:STATE ON` sets the analyzer to automatically set the vertical scale and position of the graph.

DISPlay:NOISe:GAIN:Y:PDIVision

Sets or queries the distance between graticule lines on the Gain display in dB.

Conditions Measurement view: Gain

Group Display commands

Syntax
```
DISPlay:NOISe:GAIN:Y:PDIVision <NRf>
DISPlay:NOISe:GAIN:Y:PDIVision?
```

Related Commands

Arguments <NRf> is the distance between graticule lines on the Gain display.

Returns <NR2> = a floating point value without exponents, which is the distance between graticule lines on the display in dB.

Examples `DISPLAY:NOISE:GAIN:Y:PDIVISION?` might return 1.5000000000, indicating that the distance between graticule lines on the display is 1.50 dB.

DISPlay:NOISe:GAIN:Y:POSition

Sets or queries the vertical position of the Gain display.

Conditions Measurement view: Gain

Group Display commands

Syntax
```
DISPlay:NOISe:GAIN:Y:POSition <NRf>
DISPlay:NOISe:GAIN:Y:POSition?
```

Related Commands

Arguments <NRf> is the vertical position of the Gain display.

Range: 5.12 to 26.5

Returns <NR2> = a floating point value without exponents, which is the vertical position value.

Examples `DISPLAY:NOISE:GAIN:Y:POSITION?` might return 26.5000000000, indicating that the vertical position is 26.5.

DISPlay:NOISe:GAIN:Y:SCALe

Sets or queries the vertical scale of the Gain display.

Conditions Measurement view: Gain

Group Display commands

Syntax

```
DISPlay:NOISe:GAIN:Y:SCALe <NRf>
DISPlay:NOISe:GAIN:Y:SCALe?
```

Related Commands

Arguments <NRf> is the vertical scale of the Gain display.

Range: 10.2 to 31.6

Returns <NR2> = a floating point value without an exponent, which is the vertical scale value.

Examples `DISPLAY:NOISE:GAIN:Y:SCALE?` might return 10.0000000000, indicating that the vertical scale is 10.0 dB.

DISPlay:NOISe:MEASview:DELete (No Query Form)

Deletes the selected Noise and Gain measurement view.

Conditions Measurement views: All Noise and Gain displays

Group Display commands

Syntax `DISPlay:NOISe:MEASview:DELete { GAIN | YFACtor | FIGure | TEMPerature | TABLe | UNCertainty }`

Related Commands DISPlay:NOISe:MEASview:SELect

DISPlay:NOISe:MEASview:NEW

Arguments `GAIN`: Deletes the Gain display from the view.

`YFACtor`: Deletes the Y Factor display from the view.

`FIGure`: Deletes the Noise Figure display from the view.

`TEMPerature`: Deletes the Noise Temperature display from the view.

`TABLe`: Deletes the Noise Table display from the view.

`UNCertainty`: Deletes the Uncertainty Calculator display from the view.

Examples `DISPLAY:NOISE:MEASVIEW:DELETE TABL` will delete the Noise Table display view from the screen.

DISPlay:NOISe:MEASview:NEW (No Query Form)

Displays a new Noise and Gain measurement view.

Conditions Measurement views: All Noise and Gain displays

Group Display commands

Syntax `DISPlay:NOISe:MEASview:NEW { GAIN | YFACtor | FIGure | TEMPerature | TABLe | UNCertainty }`

Related Commands DISPlay:NOISe:MEASview:SELect

DISPlay:NOISe:MEASview:DELete

Arguments `GAIN`: Displays the Gain display.

`YFACtor`: Displays the Y Factor display.

`FIGure`: Displays the Noise Figure display.

TEMPerature: Displays the Noise Temperature display.

TABLe: Displays the Noise Table display.

UNCertainty: Displays the Uncertainty Calculator display.

Examples DISPLAY:NOISE:MEASVIEW:NEW YFAC will bring the Y Factor display up on the screen.

DISPlay:NOISe:MEASview:SELect

Selects a noise measurement view. The query command returns the currently selected view.

Selecting a measurement optimizes it. Other measurements may be optimized as a side effect. Refer to the DISplay:WINDow:OPTImized:MEASurement query.

Conditions Measurement views: All Noise and Gain displays

The measurement view must be displayed.

Group Display commands

Syntax
```
DISPlay:NOISe:MEASview:SELect { GAIN | YFACtor | FIGure |
TEMPerature | TABLe | UNCertainty }
DISPlay:NOISe:MEASview:SELect?
```

Related Commands DISPlay:NOISe:MEASview:DELete

DISPlay:NOISe:MEASview:NEW

Arguments GAIN: Selects the Gain display.

YFACtor: Selects the Y Factor display.

FIGure: Selects the Noise Figure display.

TEMPerature: Selects the Noise Temperature display.

TABLe: Selects the Noise Table display.

UNCertainty: Selects the Uncertainty Calculator display.

Returns GAIN: Means that the Gain display is selected.

YFAC: Means that the Y Factor display is selected.

FIG: Means that the Noise Figure display is selected.

TEMP: Means that the Noise Temperature display is selected.

TABL: Means that the Noise Table display is selected.

UNC: Means that the Uncertainty Calculator display is selected.

Examples DISPLAY:NOISE:MEASVIEW:SELECT GAIN will select the Gain display.

DISPlay:NOISe:POWer:LINear:STATe

Sets or queries whether the linear units are shown (on) or not shown (off) on the Noise Table display.

Conditions Measurement view: Noise Table

Group Display commands

Syntax
```
DISPlay:NOISe:POWer:LINear:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:POWer:LINear:STATe?
```

Arguments OFF or 0 sets the analyzer to show the linear units on the display.

ON or 1 sets the analyzer to not show the linear units on the display.

Returns OFF or 0 indicates that the analyzer is set to show the linear units on the display.

ON or 1 indicates that the analyzer is set to not show the linear units on the display.

Examples DISPLAY:NOISE:POWER:LINEAR:STATE? might return 1, indicating that linear units are not showing on the Noise Table display.

DISPlay:NOISe:TEMPerature[:SCALe]:AUTO (No Query Form)

Automatically scales the Noise Temperature graph.

Conditions Measurement view: Noise Temperature

Group Display commands

Syntax `DISPlay:NOISe:TEMPerature[:SCALe]:AUTO`

Arguments None

DISPlay:NOISe:TEMPerature:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries the graticule state in the Noise Temperature display to on (showing) or off (hidden).

Conditions Measurement view: Noise Temperature

Group Display commands

Syntax
```
DISPlay:NOISe:TEMPerature:WINDow:TRACe:GRATicule:GRID:STATe
{ OFF | ON | 0 | 1 }
DISPlay:NOISe:TEMPerature:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule in the Noise Temperature display.

`ON` or `1` shows the graticule in the Noise Temperature display.

Returns `0` means the graticule in the Noise Temperature display is hidden (off).

`1` means the graticule in the Noise Temperature display is showing (on).

Examples `DISPLAY:NOISE:TEMPERATURE:WINDOW:TRACE:GRATICULE:GRID:STATE ON` will turn on the graticule in the Noise Temperature display.

`DISPLAY:NOISE:TEMPERATURE:WINDOW:TRACE:GRATICULE:GRID:STATE?` might return 1, indicating that the graticule in the Noise Temperature display is showing (on).

DISPlay:NOISe:TEMPerature:WINDow:TRACe:LEGend:STATe

Sets or queries the trace legend state in the Noise Temperature display to on (showing) or off (hidden).

Conditions Measurement view: Noise Temperature

Group Display commands

Syntax

```
DISPlay:NOISe:TEMPerature:WINDow:TRACe:LEGend:STATe { OFF |
ON | 0 | 1 }
DISPlay:NOISe:TEMPerature:WINDow:TRACe:LEGend:STATe?
```

Arguments

`OFF` or `0` hides the trace legend in the Noise Temperature display.

`ON` or `1` shows the trace legend in the Noise Temperature display.

Returns

`0` means the trace legend in the Noise Temperature display is hidden (off).

`1` means the trace legend in the Noise Temperature display is showing (on).

Examples

`DISPLAY:NOISE:TEMPERATURE:WINDOW:TRACE:LEGEND:STATE ON` will turn on the trace legend in the Noise Temperature display.

`DISPLAY:NOISE:TEMPERATURE:WINDOW:TRACE:LEGEND:STATE? 1` indicates that the trace legend in the Noise Temperature display is showing (on).

DISPlay:NOISe:TEMPerature:WINDow:TRACe:MEASPoints:STATe

Sets or queries the measurement points state in the Noise Temperature display to on (showing) or off (hidden).

Conditions

Measurement view: Noise Temperature

Group

Display commands

Syntax

```
DISPlay:NOISe:TEMPerature:WINDow:TRACe:MEASPoints:STATe {
OFF | ON | 0 | 1 }
DISPlay:NOISe:TEMPerature:WINDow:TRACe:MEASPoints:STATe?
```

Arguments

`OFF` or `0` hides the measurement points in the Noise Temperature display.

`ON` or `1` shows the measurement points in the Noise Temperature display.

Returns

`0` means the measurement points in the Noise Temperature display is hidden (off).

`1` means the measurement points in the Noise Temperature display is showing (on).

Examples

`DISPLAY:NOISE:TEMPERATURE:WINDOW:TRACE:MEASPOINTS:STATE ON` will turn on the measurement points in the Noise Temperature display.

DISPLAY:NOISE:TEMPERATURE:WINDOW:TRACE:MEASPOINTS:STATE? might return 1, indicating that the measurement points in the Noise Temperature display are showing (on).

DISPlay:NOISe:TEMPerature:X:AUTO (No Query Form)

Automatically sets the horizontal scale of the Noise Temperature graph.

Conditions Measurement view: Noise Temperature

Group Display commands

Syntax
```
DISPlay:NOISe:TEMPerature:X:AUTO
```

Arguments None

DISPlay:NOISe:TEMPerature:X:OFFSet

Sets or queries the center frequency (offset) value for the Noise Temperature display.

Conditions Measurement view: Noise Temperature

Group Display commands

Syntax
```
DISPlay:NOISe:TEMPerature:X:OFFSet <NRf>
DISPlay:NOISe:TEMPerature:X:OFFSet?
```

Arguments <NRf> is the center frequency (offset) value.

Range: -1.9900 GHz to +5.0000 GHz

Returns <NR3> = a floating point value with an exponent, which is the center frequency (offset) value.

Examples DISPLAY:NOISE:TEMPERATURE:X:OFFSET? might return 2.0000000009E+9, indicating that the center frequency (offset) value is 2.0000 GHz.

DISPlay:NOISe:TEMPerature:X[:SCALe]

Sets or queries the horizontal scale of the Noise Temperature display.

Conditions Measurement view: Noise Temperature

Group Display commands

Syntax

```
DISPlay:NOISe:TEMPerature:X[:SCALe] <NRf>
DISPlay:NOISe:TEMPerature:X[:SCALe]
```

Related Commands DISPlay:NOISe:GAIN:Y:SCALe

Arguments <NRf> is the horizontal scale of the Noise Temperature display.

Range: 100 Hz to 6.2 GHz

Returns <NR3> = a floating point value with an exponent, which is the horizontal scale value.

Examples `DISP:NOIS:TEMP:X:SCAL?` might return `2.9900000000E+9`, indicating that the horizontal scale value is 3.0 GHz.

DISPlay:NOISe:TEMPerature:X:STARt

Sets or queries the Start Frequency value in the Noise Temperature display.

Conditions Measurement views: Noise Temperature

Group Display commands

Syntax

```
DISPlay:NOISe:TEMPerature:X:STARt <NRf>
DISPlay:NOISe:TEMPerature:X:STARt?
```

Arguments <NRf> is the Start Frequency value.

Returns <NR3> = floating point value with an exponent, which is the Start Frequency value.

Examples

`DISPLAY:NOISE:TEMPERATURE:X:START 1.5 GHz` sets the Start Frequency in the Noise Temperature display to 1.5000 GHz.

`DISPLAY:NOISE:TEMPERATURE:X:START?` might return 1.5000000000E+9, indicating that the Start Frequency in the Noise Temperature display is 1.5000 GHz.

DISPlay:NOISe:TEMPerature:X:STOP

Sets or queries the Stop Frequency value in the Noise Temperature display.

Conditions

Measurement views: Noise Temperature

Group

Display commands

Syntax

```
DISPlay:NOISe:TEMPerature:X:STOP <NRf>
DISPlay:NOISe:TEMPerature:X:STOP?
```

Arguments

`<NRf>` is the Stop Frequency value.

Returns

`<NR3>` = floating point value with an exponent, which is the Stop Frequency value.

Examples

`DISPLAY:NOISE:TEMPERATURE:X:STOP 1.5 GHz` sets the Stop Frequency in the Noise Temperature display to 1.5000 GHz.

`DISPLAY:NOISE:TEMPERATURE:X:STOP?` might return 1.5000000000E+9, indicating that the Stop Frequency in the Noise Temperature display is 1.5000 GHz.

DISPlay:NOISe:TEMPerature:Y:AUTO (No Query Form)

Automatically sets the vertical scale of the Noise Temperature graph.

Conditions

Measurement view: Noise Temperature

Group

Display commands

Syntax

```
DISPlay:NOISe:TEMPerature:Y:AUTO
```

Arguments None

DISPlay:NOISe:TEMPerature:Y:AUTO:STATe

Sets or queries the state of the automatic vertical scale and position of the graph to on or off.

Conditions Measurement view: Noise Temperature

Group Display commands

Syntax

```
DISPlay:NOISe:TEMPerature:Y:AUTO:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:TEMPerature:Y:AUTO:STATe?
```

Arguments `OFF` or `0` sets the analyzer to manual setting of the vertical scale and position of the graph.

`ON` or `1` sets the analyzer to automatically set the vertical scale and position of the graph.

Returns `OFF` or `0` means the analyzer is in manual mode for setting of the vertical scale and position of the graph.

`ON` or `1` means the analyzer is in automatic mode for setting of the vertical scale and position of the graph.

Examples `DISPLAY:NOISE:TEMPERATURE:Y:AUTO:STATE ON` sets the analyzer to automatically set the vertical scale and position of the graph.

DISPlay:NOISe:TEMPerature:Y:PDIVision

Sets or queries the distance between graticule lines on the display in dB.

Conditions Measurement view: Noise Temperature

Group Display commands

Syntax

```
DISPlay:NOISe:TEMPerature:Y:PDIVision <NRf>
DISPlay:NOISe:TEMPerature:Y:PDIVision?
```

Arguments <NRf> is the distance between graticule lines on the display.

Range: 100 mdB to 5.00 dB

Returns <NR2> = a floating point value without an exponent, which is the distance between graticule lines on the display in dB.

Examples `DISPLAY:NOISE:TEMPERATURE:Y:PDIVISION?` might return 1.5000000000, indicating that the distance between graticule lines on the display is 1.50 dB.

DISPlay:NOISe:TEMPerature:Y:POSition

Sets or queries the vertical position of the Noise Temperature display.

Conditions Measurement view: Noise Temperature

Group Display commands

Syntax

```
DISPlay:NOISe:TEMPerature:Y:POSition <NRf>
DISPlay:NOISe:TEMPerature:Y:POSition?
```

Arguments <NRf> is the vertical position of the Noise Temperature display.

Range: 5.12 to 26.5

Returns <NR2> = a floating point value without an exponent, which is the vertical position value.

The vertical position value

Examples `DISPLAY:NOISE:TEMPERATURE:Y:POSITION?` might return 26.5000000000, indicating that the vertical position is 26.5.

DISPlay:NOISe:TEMPerature:Y:SCALe

Sets or queries the vertical scale of the Noise Temperature display.

Conditions Measurement view: Noise Temperature

Group Display commands

Syntax
```
DISPlay:NOISe:TEMPerature:Y:SCALe <NRf>
DISPlay:NOISe:TEMPerature:Y:SCALe?
```

Arguments <NRf> is the vertical scale of the Noise Temperature display.

Range: 10.2 to 31.6

Returns <NR2> = a floating point value without an exponent, which is the vertical scale value.

Examples `DISPLAY:NOISE:TEMPERATURE:Y:SCALE?` might return 10.0000000000, indicating that the vertical scale is 10.0 dB.

DISPlay:NOISe:UNCERTainty:RESult:GAIN? (Query Only)

Queries the gain computed uncertainty value (dB).

Conditions Measurement view: Uncertainty Calculator

Group Display commands

Syntax
```
DISPlay:NOISe:UNCERTainty:RESult:GAIN?
```

Related Commands DISPlay:NOISe:UNCERTainty:RESult:NFIGure?

Returns <NR3> = a floating point value with an exponent, which is the gain computed uncertainty value (dB).

Examples `DISPLAY:NOISE:UNCERTAINTY:RESULT:GAIN?` might return 783.0000000000E-3, indicating that the computed uncertainty value for gain is 0.783 dB.

DISPlay:NOISe:UNCERTainty:RESult:NFIGure? (Query Only)

Queries the noise figure computed uncertainty value (dB).

Conditions Measurement view: Uncertainty Calculator

Group Display commands

Syntax `DISPlay:NOISe:UNCERTainty:RESult:NFIGure?`

Related Commands DISPlay:NOISe:UNCERTainty:RESult:GAIN?

Returns <NR3> = a floating point value with an exponent, which is the noise figure computed uncertainty value (dB).

Examples `DISPLAY:NOISE:UNCERTAINTY:RESULT:NFIGURE?` might return 783.0000000000E-3, indicating that the computed uncertainty value for noise figure is 0.783 dB.

DISPlay:NOISe:YFACtor[:SCALe]:AUTO (No Query Form)

Automatically scales the Y Factor graph.

Conditions Measurement view: Y Factor

Group Display commands

Syntax `DISPlay:NOISe:YFACtor[:SCALe]:AUTO`

Arguments None

DISPlay:NOISe:YFACtor:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries the graticule state in the Y Factor display to on (showing) or off (hidden).

Conditions Measurement view: Y Factor

Group Display commands

Syntax

```
DISPlay:NOISe:YFACtor:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:YFACtor:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or `0` hides the graticule in the Y Factor display.

`ON` or `1` shows the graticule in the Y Factor display.

Returns

`0` means the graticule in the Y Factor display is hidden (off).

`1` means the graticule in the Y Factor display is showing (on).

Examples

`DISPLAY:NOISE:YFACTOR:WINDOW:TRACE:GRATICULE:GRID:STATE ON` will turn on the graticule in the Y Factor display.

`DISPLAY:NOISE:YFACTOR:WINDOW:TRACE:GRATICULE:GRID:STATE?` might return 1, indicating that the graticule in the Y Factor display is showing (on).

DISPlay:NOISe:YFACtor:WINDow:TRACe:LEGend:STATe

Sets or queries the trace legend state in the Y Factor display to on (showing) or off (hidden).

Conditions

Measurement view: Y Factor

Group

Display commands

Syntax

```
DISPlay:NOISe:YFACtor:WINDow:TRACe:LEGend:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:YFACtor:WINDow:TRACe:LEGend:STATe?
```

Arguments

`OFF` or `0` hides the trace legend in the Y Factor display.

`ON` or `1` shows the trace legend in the Y Factore display.

Returns

`0` means the trace legend in the Y Factor display is hidden (off).

`1` means the trace legend in the Y Factor display is showing (on).

Examples

`DISPLAY:NOISE:YFACTOR:WINDOW:TRACE:LEGEND:STATE ON` will turn on the trace legend in the Y Factor display.

DISPLAY:NOISE:YFACTOR:WINDOW:TRACE:LEGEND:STATE? 1 indicates that the trace legend in the Y Factor display is showing (on).

DISPlay:NOISe:YFACtor:WINDow:TRACe:MEASPoints:STATe

Sets or queries the measurement points state in the Y Factor display to on (showing) or off (hidden).

Conditions Measurement view: Y Factor

Group Display commands

Syntax

```
DISPlay:NOISe:YFACtor:WINDow:TRACe:MEASPoints:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:YFACtor:WINDow:TRACe:MEASPoints:STATe?
```

Arguments OFF or 0 hides the measurement points in the Y Factor display.

ON or 1 shows the measurement points in the Y Factor display.

Returns 0 means the measurement points in the Y Factor display is hidden (off).

1 means the measurement points in the Y Factor display is showing (on).

Examples DISPLAY:NOISE:YFACTOR:WINDOW:TRACE:MEASPOINTS:STATE ON will turn on the measurement points in the Y Factor display.

DISPLAY:NOISE:YFACTOR:WINDOW:TRACE:MEASPOINTS:STATE? might return 1, indicating that the measurement points in the Y Factor display are showing (on).

DISPlay:NOISe:YFACtor:X:AUTO (No Query Form)

Automatically sets the horizontal scale of the Y Factor graph.

Conditions Measurement view: Y Factor

Group Display commands

Syntax

```
DISPlay:NOISe:YFACtor:X:AUTO
```

Arguments None

DISPlay:NOISe:YFACtor:X:OFFSet

Sets or queries the center frequency (offset) in the Y Factor display.

Conditions Measurement view: Y Factor

Group Display commands

Syntax
```
DISPlay:NOISe:YFACtor:X:OFFSet <NRf>
DISPlay:NOISe:YFACtor:X:OFFSet?
```

Arguments <NRf> is the center frequency (offset) in the Y Factor display.

Range: -1.9900 GHz to +5.0000 GHz

Returns <NR3> = a floating point value with an exponent, which is the center frequency (offset) value.

Examples `DISPLAY:NOISE:YFACTOR:X:OFFSET?` might return `2.0000000009E+9`, indicating that the center frequency (offset) value is 2.0000 GHz.

DISPlay:NOISe:YFACtor:X[:SCALe]

Sets or queries the horizontal scale of the Y Factor display.

Conditions Measurement view: Y Factor

Group Display commands

Syntax
```
DISPlay:NOISe:YFACtor:X[:SCALe] <NRf>
DISPlay:NOISe:YFACtor:X[:SCALe]
```

Related Commands DISPlay:NOISe:GAIN:Y:SCALe

Arguments <NRf> is the horizontal scale of the Y Factor display.

Range: 100 Hz to 6.2 GHz

Returns <NR3> = a floating point value with an exponent, which is the horizontal scale value.

Examples `DISP:NOIS:YFAC:X:SCAL?` might return `2.9900000000E+9`, indicating that the horizontal scale value is 3.0 GHz.

DISPlay:NOISe:YFACtor:X:STARt

Sets or queries the Start Frequency value in the Y Factor display.

Conditions Measurement views: Y Factor

Group Display commands

Syntax
```
DISPlay:NOISe:YFACtor:X:STARt <NRf>
DISPlay:NOISe:YFACtor:X:STARt?
```

Arguments `<NRf>` is the Start Frequency value in the Y Factor display.

Returns `<NR3>` = floating point value with an exponent, which is the Start Frequency value.

Examples `DISPLAY:NOISE:YFACTOR:X:START 1.5 GHz` sets the Start Frequency in the Y Factor display to 1.5000 GHz.

`DISPLAY:NOISE:YFACTOR:X:START?` might return 1.5000000000E+9, indicating that the Start Frequency in the Y Factor display is 1.5000 GHz.

DISPlay:NOISe:YFACtor:X:STOP

Sets or queries the Stop Frequency value Y Factor display.

Conditions Measurement views: Y Factor

Group Display commands

Syntax

```
DISPlay:NOISe:YFACtor:X:STOP <NRf>
DISPlay:NOISe:YFACtor:X:STOP?
```

Arguments

<NRf> is the Stop Frequency value in the Y Factor display.

Returns

<NR3> = floating point value with an exponent, which is the Stop Frequency value.

Examples

`DISPLAY:NOISE:YFACTOR:X:STOP 1.5 GHz` sets the Stop Frequency in the Y Factor display to 1.5000 GHz.

`DISPLAY:NOISE:YFACTOR:X:STOP?` might return 1.5000000000E+9, indicating that the Stop Frequency in the Y Factor display is 1.5000 GHz.

DISPlay:NOISe:YFACtor:Y:AUTO (No Query Form)

Automatically sets the vertical scale of the Y Factor graph.

Conditions

Measurement view: Y Factor

Group

Display commands

Syntax

```
DISPlay:NOISe:YFACtor:Y:AUTO
```

Arguments

None

DISPlay:NOISe:YFACtor:Y:AUTO:STATe

Sets or queries the state of the automatic vertical scale and position of the Y Factor graph to on or off.

Conditions

Measurement view: Y Factor

Group

Display commands

Syntax

```
DISPlay:NOISe:YFACtor:Y:AUTO:STATe { OFF | ON | 0 | 1 }
DISPlay:NOISe:YFACtor:Y:AUTO:STATe?
```

Arguments

`OFF` or `0` sets the analyzer to manual setting of the vertical scale and position of the graph.

`ON` or `1` sets the analyzer to automatically set the vertical scale and position of the graph.

Returns

`OFF` or `0` means the analyzer is in manual mode for setting of the vertical scale and position of the graph.

`ON` or `1` means the analyzer is in automatic mode for setting of the vertical scale and position of the graph.

Examples

`DISPLAY:NOISE:YFACTOR:Y:AUTO:STATE ON` sets the analyzer to automatically set the vertical scale and position of the graph.

DISPlay:NOISe:YFACtor:Y:PDIVision

Sets or queries the distance between graticule lines on the display in dB.

Conditions

Measurement view: Y Factor

Group

Display commands

Syntax

```
DISPlay:NOISe:YFACtor:Y:PDIVision <NRf>
DISPlay:NOISe:YFACtor:Y:PDIVision?
```

Arguments

<NRf> is the distance between graticule lines on the display in dB.

Range: 100 mdB to 5.00 dB

Returns

<NR2> = a floating point value without an exponent, which is the distance between graticule lines on the display in dB.

Examples

`DISPLAY:NOISE:YFACTOR:Y:PDIVISION?` might return 1.5000000000, indicating that the distance between graticule lines on the display is 1.50 dB.

DISPlay:NOISe:YFACtor:Y:POSition

Sets or queries the vertical position of the Y Factor display.

Conditions Measurement view: Y Factor

Group Display commands

Syntax
```
DISPlay:NOISe:YFACtor:Y:POSition <NRf>
DISPlay:NOISe:YFACtor:Y:POSition?
```

Arguments <NRf> is the vertical position of the Y Factor display.

Range: 5.12 to 26.5

Returns <NR2> = a floating point value without an exponent, which is the vertical position value.

Examples `DISPLAY:NOISE:YFACTOR:Y:POSITION?` might return 26.5000000000, indicating that the vertical position is 26.5.

DISPlay:NOISe:YFACtor:Y:SCALe

Sets or queries the vertical scale of the Y Factor display.

Conditions Measurement view: Y Factor

Group Display commands

Syntax
```
DISPlay:NOISe:YFACtor:Y:SCALe <NRf>
DISPlay:NOISe:YFACtor:Y:SCALe?
```

Related Commands

Arguments <NRf> is the vertical scale of the Y Factor display.

Range: 10.2 to 31.6 dB

Returns <NR2> = a floating point value without an exponent, which is the vertical scale value.

Examples `DISPLAY:NOISE:YFACTOR:Y:SCALE?` might return 10.0000000000, indicating that the vertical scale is 10.0 dB.

DISPlay:OBWidth:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the Occupied Bandwidth view.

Conditions Measurement views: Occupied Bandwidth

Group Display commands

Syntax

```
DISPlay:OBWidth:MARKer:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:OBWidth:MARKer:SHOW:STATe?
```

Arguments `OFF` or `0` hides the readout for the selected marker in the view.

`ON` or `1` shows the readout for the selected marker in the view.

Examples `DISPLAY:OBWIDTH:MARKER:SHOW:STATE ON` shows the readout for the selected marker in the view.

DISPlay:OBWidth:RESet:SCALe (No Query Form)

Resets the horizontal and vertical scale to the default values described below in the Occupied Bandwidth view.

Vertical offset = Reference level,
Vertical scale = 100 dB,
Horizontal offset = Center frequency, and
Horizontal scale = Default span

Conditions Measurement views: Occupied Bandwidth

Group Display commands

Syntax

```
DISPlay:OBWidth:RESet:SCALe
```

Arguments None

Examples `DISPLAY:OBWIDTH:RESET:SCALE` resets the horizontal and vertical scale to the default values.

DISPlay:OBWidth:SELected:BANDwidth

Sets or queries the bandwidth (OBW or x dB BW) to measure in the Occupied Bandwidth view.

Conditions Measurement views: Occupied Bandwidth

Group Display commands

Syntax
```
DISPlay:OBWidth:SELected:BANDwidth { OBWidth | XDBbandwidth }
DISPlay:OBWidth:SELected:BANDwidth?
```

Arguments `OBWidth` selects the occupied bandwidth to measure.

`XDBbandwidth` selects the x dB bandwidth to measure.

Examples `DISPLAY:OBWIDTH:SELECTED:BANDWIDTHOBWidth` selects the occupied bandwidth to measure.

DISPlay:OBWidth:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions Measurement views: Occupied Bandwidth

Group Display commands

Syntax
```
DISPlay:OBWidth:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:OBWidth:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples `DISPLAY:OBWIDTH:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the screen.

DISPlay:OBWidth:X[:SCALe]

Sets or queries the horizontal range of the Occupied Bandwidth view.

Conditions Measurement views: Occupied Bandwidth

Group Display commands

Syntax
```
DISPlay:OBWidth:X[:SCALe] <value>
DISPlay:OBWidth:X[:SCALe]?
```

Related Commands DISPlay:OBWidth:X[:SCALe]:OFFSet

Arguments `<value> ::` `<NRf>` specifies the horizontal range.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `DISPLAY:OBWIDTH:X:SCALE 10MHz` sets the horizontal range to 10 MHz.

DISPlay:OBWidth:X[:SCALe]:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the Occupied Bandwidth view.

Conditions Measurement views: Occupied Bandwidth

Group Display commands

Syntax
```
DISPlay:OBWidth:X[:SCALe]:AUTO
```

Arguments None

Examples `DISPLAY:OBWIDTH:X:SCALE:AUTO` rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:OBWidth:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (left edge) of the Occupied Bandwidth view.

Conditions Measurement views: Occupied Bandwidth

Group Display commands

Syntax
```
DISPlay:OBWidth:X[:SCALe]:OFFSet <value>
DISPlay:OBWidth:X[:SCALe]:OFFSet?
```

Related Commands DISPlay:OBWidth:X[:SCALe]

Arguments `<value> ::= <NRf>` specifies the minimum horizontal value.
Range: [(center frequency) - (X scale) × 0.9] to [(center frequency) + (X scale) × 0.9]

Examples `DISPLAY:OBWIDTH:X:SCALE:OFFSET 1.45GHz` sets the minimum horizontal value to 1.45 GHz in the Occupied Bandwidth view.

DISPlay:OBWidth:Y[:SCALe]

Sets or queries the vertical range of the Occupied Bandwidth view.

Conditions Measurement views: Occupied Bandwidth

Group Display commands

Syntax
```
DISPlay:OBWidth:Y[:SCALe] <value>
DISPlay:OBWidth:Y[:SCALe]?
```

Related Commands DISPlay:OBWidth:Y[:SCALe]:OFFSet

Arguments `<value> ::` `<NRf>` specifies the vertical range. Range: 0.1 to 200 dB.

Examples `DISPLAY:OBWIDTH:Y:SCALE 100` sets the vertical range to 100 dB in the Occupied Bandwidth view.

DISPlay:OBWidth:Y[:SCALe]:AUTO (No Query Form)

Rescales the vertical axis automatically to fit the waveform to the screen in the Occupied Bandwidth view.

Conditions Measurement views: Occupied Bandwidth

Group Display commands

Syntax
```
DISPlay:OBWidth:Y[:SCALe]:AUTO
```

Arguments None

Examples `DISPLAY:OBWIDTH:Y:SCALE:AUTO` rescales the vertical scale automatically to fit the waveform to the screen.

DISPlay:OBWidth:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the top edge of the vertical axis) in the Occupied Bandwidth view.

Conditions Measurement views: Occupied Bandwidth

Group Display commands

Syntax
```
DISPlay:OBWidth:Y[:SCALe]:OFFSet <value>
DISPlay:OBWidth:Y[:SCALe]:OFFSet?
```

Related Commands DISPlay:OBWidth:Y[:SCALe]

Arguments `<value> ::` `<NRf>` specifies the vertical offset. Range: -170 to +50 dBm.

Examples `DISPLAY:OBWIDTH:Y:SCALE:OFFSET -12.5` sets the vertical offset to -12.5 dBm in the Occupied Bandwidth view.

DISPlay:OFDM:CONSte[:SCALe]

Sets or queries the scale (zoom) in the OFDM Constellation view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:CONSte[:SCALe] <value>
DISPlay:OFDM:CONSte[:SCALe]?
```

Arguments `<value> ::` `<NRf>` specifies the zoom setting in the OFDM Constellation view.

A positive value zooms out and a negative value zooms in.

Examples `DISPLAY:OFDM:CONSTE[:SCALE] 5` zooms the display out by 5

DISPlay:OFDM:CONSte:X:OFFSet

Sets or queries the horizontal offset in the OFDM Constellation view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:CONSte:X:OFFSet <value>
DISPlay:OFDM:CONSte:X:OFFSet?
```

Arguments `<value> ::` `<NRf>` specifies the offset range.

Examples `DISPLAY:OFDM:CONSTE:X:OFFSET 2.7` sets the horizontal offset to 2.7 units to the left.

DISPlay:OFDM:CONSte:Y:OFFSet

Sets or queries the vertical offset in the OFDM Constellation view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:CONSte:Y:OFFSet <value>
DISPlay:OFDM:CONSte:Y:OFFSet?
```

Arguments <value> :: <NRf> specifies the offset range.

A positive value moves the offset up; and negative value moves the offset down.

Examples DISPLAY:OFDM:CONSTE:Y:OFFSET 2 sets the vertical offset to up 2 units.

DISPlay:OFDM:CRESponse:FREQuency:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the OFDM Channel response graph.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:CRESponse:FREQuency:AUTO
```

Arguments None

Examples DISPLAY:OFDM:CRESPONSE:FREQUENCY:AUTO rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:OFDM:CRESponse:FREQuency:OFFSet

Sets or queries the frequency offset in the OFDM Channel response graph.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:CRESponse:FREQuency:OFFSet <value>
DISPlay:OFDM:CRESponse:FREQuency:OFFSet?
```

Arguments <value> :: <NRf> specifies the frequency offset.

Examples `DISPLAY:OFDM:CRESPONSE:FREQUENCY:OFFSET 1.45E+9` sets the frequency offset to 1.45 GHz.

DISPlay:OFDM:CRESponse:FREQuency[:SCALe]

Sets or queries the horizontal range of the OFDM Channel response graph.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:CRESponse:FREQuency[:SCALe] <value>
DISPlay:OFDM:CRESponse:FREQuency[:SCALe]?
```

Arguments <value>::= <NRf> specifies the horizontal range.

Examples `DISPLAY:OFDM:CRESPONSE:FREQUENCY[:SCALE] 10E+6` sets the horizontal range to 10 MHz.

DISPlay:OFDM:CRESponse:MAGNitude:AUTO (No Query Form)

Rescales the magnitude automatically to fit the waveform to the screen in the OFDM Channel response graph.

Conditions Measurement views: OFDM

Group Display commands

Syntax `DISPlay:OFDM:CRESponse:MAGNitude:AUTO`

Arguments None

Examples `DISPLAY:OFDM:CRESPONSE:MAGNITUDE:AUTO` rescales the magnitude scale automatically to fit the waveform to the screen.

DISPlay:OFDM:CRESponse:MAGNitude:OFFSet

Sets or queries the magnitude offset in the OFDM Channel response graph.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:CRESponse:MAGNitude:OFFSet <value>
DISPlay:OFDM:CRESponse:MAGNitude:OFFSet?
```

Arguments <value>::= <NRf> specifies the magnitude offset.

Examples `DISPLAY:OFDM:CRESPONSE:MAGNITUDE:OFFSET 5` sets the magnitude offset to 5 dB.

DISPlay:OFDM:CRESponse:MAGNitude:PDIVision

Sets or queries the scale (per division) of the magnitude in the OFDM Channel response graph.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:CRESponse:MAGNitude:PDIVision <value>
DISPlay:OFDM:CRESponse:MAGNitude:PDIVision?
```

Arguments <value> :: <NRf> specifies the vertical scale (per division).

Examples `DISPLAY:OFDM:CRESPONSE:MAGNITUDE:PDIVISION 2.7` sets the vertical scale to 2.70 dB/div.

DISPlay:OFDM:CRESponse:PHASe:AUTO (No Query Form)

Rescales the vertical axis automatically to fit the waveform to the screen in the OFDM Channel response graph.

Conditions Measurement views: OFDM

Group Display commands

Syntax `DISPlay:OFDM:CRESponse:PHASe:AUTO`

Arguments None

Examples `DISPLAY:OFDM:CRESPONSE:PHASE:AUTO` rescales the vertical scale automatically to fit the waveform to the screen.

DISPlay:OFDM:CRESponse:PHASe:OFFSet

Sets or queries the phase offset in the OFDM Channel response graph.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:CRESponse:PHASe:OFFSet <value>
DISPlay:OFDM:CRESponse:PHASe:OFFSet?
```

Arguments <value> :: <NRf> specifies the phase offset.

Examples `DISPLAY:OFDM:CRESPONSE:PHASE:OFFSET -14.5` sets the minimum vertical value to -14.5 ° in the OFDM Channel response graph.

DISPlay:OFDM:CRESponse:PHASe[:SCALe]

Sets the vertical scale automatically to fit the waveform to the screen in the OFDM Channel response graph.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:CRESponse:PHASe[:SCALe] <value>
DISPlay:OFDM:CRESponse:PHASe[:SCALe]?
```

Arguments <value> :: <NRf> specifies the vertical range.

Examples `DISPLAY:OFDM:CRESPONSE:PHASE[:SCALE] 30` sets the vertical range to 30 ° in the OFDM Channel response graph.

DISPlay:OFDM:CRESponse:WINDow:SELect:PLOT

Sets or queries the plot in the OFDM Channel Response window measurement.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:CRESponse:WINDow:SELect:PLOT { MAGNitude |
PHASe }
DISPlay:OFDM:CRESponse:WINDow:SELect:PLOT?
```

Arguments `MAGNitude` selects the magnitude display.

`PHASe` selects the phase display

Examples `DISPLAY:OFDM:CRESPONSE:WINDOW:SELECT:PLOT PHASe` selects the phase display.

DISPlay:OFDM:CRESponse:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule in the OFDM Channel Response view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:CRESponse:WINDow:TRACe:GRATicule:GRID:STATe {
OFF | ON | 1| 0}
DISPlay:OFDM:CRESponse:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule in the graph.

`ON` or `1` shows the graticule in the graph.

Examples `DISPLAY:OFDM:CRESPONSE:WINDOW:TRACE:GRATICULE:GRID:STATE OFF` turns off the graticule in the graph.

DISPlay:OFDM:EVM:FREQuency:AUTO (No Query Form)

Rescales the frequency automatically to fit the waveform to the screen in the OFDM EVM graph.

Conditions Measurement views: OFDM

Group Display commands

Syntax `DISPlay:OFDM:EVM:FREQuency:AUTO`

Arguments None

Examples `DISPLAY:OFDM:EVM:FREQUENCY:AUTO` rescales the frequency scale automatically to fit the waveform to the screen.

DISPlay:OFDM:EVM:FREQuency:OFFSet

Sets or queries the frequency offset in the OFDM EVM graph.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:EVM:FREQuency:OFFSet <value>
DISPlay:OFDM:EVM:FREQuency:OFFSet
```

Arguments <value> :: <NRf> specifies the frequency offset.

Examples `DISPLAY:OFDM:EVM:FREQUENCY:OFFSET -21` sets the frequency offset to –21 subcarriers.

DISPlay:OFDM:EVM:FREQuency[:SCALe]

Sets or queries the horizontal range of the OFDM EVM graph.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:EVM:FREQuency[:SCALe] <value>
DISPlay:OFDM:EVM:FREQuency[:SCALe]?
```

Arguments <value>::= <NRf> specifies the horizontal range.

Examples `DISPLAY:OFDM:EVM:FREQUENCY[:SCALE] 10E+6` sets the horizontal range to 10 MHz.

DISPlay:OFDM:EVM:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the OFDM EVM view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:EVM:MARKer:SHOW:STATe { OFF | ON | 1| 0 }
DISPlay:OFDM:EVM:MARKer:SHOW:STATe?
```

Arguments `OFF` or `0` hides the readout for the selected marker in the graph.

`ON` or `1` shows the readout for the selected marker in the graph.

Examples `DISPLAY:OFDM:EVM:MARKER:SHOW:STATE ON` shows the readout for the selected marker in the graph.

DISPlay:OFDM:EVM:TIME:AUTO (No Query Form)

Rescales the time automatically to fit the waveform to the screen in the OFDM EVM view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:EVM:TIME:AUTO
```

Arguments None

Examples `DISPLAY:OFDM:EVM:TIME:AUTO` rescales the time scale automatically to fit the waveform to the screen.

DISPlay:OFDM:EVM:TIME:OFFSet

Sets or queries the time offset in the OFDM EVM view.

Conditions Measurement views: OFDM

Group Display commands

Syntax

```
DISPlay:OFDM:EVM:TIME:OFFSet <value>
DISPlay:OFDM:EVM:TIME:OFFSet?
```

Arguments

<value>::= <NRf> specifies the magnitude offset.

Examples

`DISPLAY:OFDM:EVM:TIME:OFFSET –11.7` sets the time scale offset to –11.7 symbols.

DISPlay:OFDM:EVM:TIME[:SCALe]

Sets or queries the time scale in the OFDM EVM view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:EVM:TIME[:SCALe] <value>
DISPlay:OFDM:EVM:TIME[:SCALe]?
```

Arguments

<value>::= <NRf> specifies the time scale.

Examples

`DISPLAY:OFDM:EVM:TIME[:SCALE] 13` sets the time scale to 13 symbols.

DISPlay:OFDM:EVM:WINDow:SELect:PLOT

Sets or queries the plot in the OFDM EVM window measurement.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:EVM:WINDow:SELect:PLOT { BOTH | SCARrier |
SYMBols }
DISPlay:OFDM:EVM:WINDow:SELect:PLOT?
```

Arguments

`SCARrier` selects the subcarrier display.

`SYMBols` selects the symbols display

BOTH selects both displays.

Examples DISPLAY:OFDM:EVM:WINDOW:SELECT:PLOT SYMBols selects the symbols display.

DISPlay:OFDM:EVM:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule in the OFDM EVM view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:EVM:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 1| 0}
DISPlay:OFDM:EVM:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments OFF or 0 hides the graticule in the graph.

ON or 1 shows the graticule in the graph.

Examples DISPLAY:OFDM:EVM:WINDOW:TRACE:GRATICULE:GRID:STATE OFF turns off the graticule in the graph.

DISPlay:OFDM:EVM:Y:AUTO (No Query Form)

Rescales the vertical scale automatically to fit the waveform in the OFDM EVM view.

Conditions Measurement views: OFDM

Group Display commands

Syntax DISPlay:OFDM:EVM:Y:AUTO

Arguments None

Examples `DISPLAY:OFDM:EVM:Y:AUTO` rescales the vertical scale to fit the waveform on the screen.

DISPlay:OFDM:EVM:Y:OFFSet

Sets or queries the vertical offset in the OFDM EVM view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:EVM:Y:OFFSet <value>
DISPlay:OFDM:EVM:Y:OFFSet?
```

Arguments <value>::= <NRf> specifies the vertical offset.

Examples `DISPLAY:OFDM:EVM:Y:OFFSET –20` sets the vertical offset to –20%.

DISPlay:OFDM:EVM:Y[:SCALe]

Sets or queries the vertical range of the OFDM EVM view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:EVM:Y[:SCALe] <value>
DISPlay:OFDM:EVM:Y[:SCALe]?
```

Arguments <value>::= <NRf> specifies the vertical scale.

Examples `DISPLAY:OFDM:EVM:Y[:SCALE] 200` sets the vertical scale to 200%.

DISPlay:OFDM:FLATness:AUTO (No Query Form)

Automatically rescales the horizontal and vertical axes for the best display in the OFDM Spectral Flatness view.

Conditions Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group Display commands

Syntax `DISPlay:OFDM:FLATness:AUTO`

Arguments None

Examples `DISP:OFDM:FLAT:AUTO` automatically rescales the horizontal and vertical axes for the best display.

DISPlay:OFDM:FLATness:X:AUTO (No Query Form)

Rescales the horizontal axis automatically in the OFDM Spectral Flatness display.

Conditions Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group Display commands

Syntax `DISPlay:OFDM:FLATness:X:AUTO`

Arguments None

Examples `DISP:OFDM:FLAT:X:AUTO` rescales the horizontal axis automatically for the best display.

DISPlay:OFDM:FLATness:X:OFFSet

Sets or queries the value of the offset of the horizontal axis from the center of the OFDM WLAN Spectral Flatness display.

Conditions Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group Display commands

Syntax
```
DISPlay:OFDM:FLATness:X:OFFSet <NRf>
DISPlay:OFDM:FLATness:X:OFFSet?
```

Arguments Floating point number between limits which are set as a function of the current scale value guaranteed to keep the plot at least partially visible on the display. Positive inputs push the plot down (as viewed by the user) and negative inputs push the plot up. 0 recenters the plot.

Returns The value of the offset of the X axis from the center of the plot as a floating point number.

Examples `DISP:OFDM:FLAT:X:OFFS 150` shifts the display to the left by 150%.

DISPlay:OFDM:FLATness:X[:SCALe]

Sets or queries the value of the horizontal scale in the OFDM Spectral Flatness display. The units are Frequency or Subcarrier. To set the units, use the command `[SENSe]:OFDM:UNIT:FREQuency`.

Conditions Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group Display commands

Syntax
```
DISPlay:OFDM:FLATness:X[:SCALe] <NRf>
DISPlay:OFDM:FLATness:X[:SCALe]?
```

Arguments Floating point number that represents the value of the horizontal scale.

Examples `DISP:OFDM:FLAT:X:SCALE 32` sets the horizontal scale to 32.

DISPlay:OFDM:FLATness:Y:AUTO (No Query Form)

Rescales the vertical axis scale and position values automatically in the OFDM Spectral Flatness display.

Conditions Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group Display commands

Syntax `DISPlay:OFDM:FLATness:Y:AUTO`

Arguments None

Examples `DISP:OFDM:FLAT:Y:AUTO` automatically selects the vertical scale and position values.

DISPlay:OFDM:FLATness:Y:OFFSet

Sets or queries the value of the offset from the center (vertical position), in percent terms, for the OFDM Spectral Flatness display. To set the Vertical Scale, use the command DISPlay:OFDM:FLATness:Y[:SCALe]

Conditions Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group Display commands

Syntax
```
DISPlay:OFDM:FLATness:Y:OFFSet <NRf>
DISPlay:OFDM:FLATness:Y:OFFSet?
```

Arguments Floating point number between limits which are set as a function of the current scale value guaranteed to keep the plot at least partially visible on the display. Positive inputs push the plot down (as viewed by the user) and negative inputs push the plot up. 0 re-centers the plot.

Returns The value of the offset from the center of the Y axis for the plot as a floating point number.

Examples `DISP:OFDM:FLAT:Y:OFFS 150` sets the offset from the center (vertical position) of the plot to 150%.

DISPlay:OFDM:FLATness:Y[:SCALe]

Sets or queries the vertical scale value for the OFDM Spectral Flatness display. To set the vertical position (offset) value, use the command DISPlay:OFDM:FLATness:Y:OFFSet.

Conditions Measurement view: OFDM

This command requires Option 22, "OFDM Measurements".

Group Display commands

Syntax

```
DISPlay:OFDM:FLATness:Y[:SCALe] <NRf>
DISPlay:OFDM:FLATness:Y[:SCALe]?
```

Arguments Floating point number that specifies the vertical scale value.

Examples `DISP:OFDM:FLAT:Y:SCALE 10` sets the vertical scale value to 10.

DISPlay:OFDM:MEASview:DELete (No Query Form)

Deletes the specified OFDM view.

Conditions Measurement views: OFDM

Group Display commands

Syntax

```
DISPlay:OFDM:MEASview:DELete { CONSte | SUMMary | STABle |
CRESponse | EVM | PERRor | MERRor | POWer }
```

Arguments CONSte deletes the OFDM Constellation view.

SUMMary deletes the OFDM Summary view.

STABle deletes the OFDM Symbol table view.

CRESponse deletes the OFDM Channel response view.

EVM deletes the EVM (Error Vector Magnitude) versus Time view.

PERRor deletes the Phase error versus Time view.

MERRor deletes the Magnitude error versus Time view.

POWer deletes the OFDM Power view.

If you attempt to delete a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running") will be returned.

Examples `DISPLAY:OFDM:MEASVIEW:DELETE CONSte` deletes the OFDM Constellation view.

DISPlay:OFDM:MEASview:NEW (No Query Form)

Displays a new OFDM measurement view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:MEASview:NEW { CONSte | SUMMary | STABle |
CRESponse | EVM | PERRor | MERRor | POWer }
```

Arguments CONSte creates a new OFDM Constellation view.

SUMMary creates a new OFDM Summary view.

STABle creates a new OFDM Symbol table view.

CRESponse creates a new OFDM Channel response view.

EVM creates a new EVM (Error Vector Magnitude) versus Time view.

PERRor creates a new Phase error versus Time view.

MERRor creates a new Magnitude error versus Time view.

POWer creates a new OFDM Power view.

If you attempt to open a view that is currently displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Examples `DISPLAY:OFDM:MEASVIEW:NEW`:CONSte creates a new OFDM Constellation view.

DISPlay:OFDM:MEASview:SELect

Selects a OFDM measurement view on the screen. The query returns the currently selected view.

Selecting a measurement optimizes it. Other measurements may be optimized as a side effect. Refer to the DISPlay:WINDow:OPTimized:MEASurement? query.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:MEASview:SELect { CONSte | SUMMary | STABle |
CRESponse | EVM | PERRor | MERRor | POWer }
DISPlay:OFDM:MEASview:SELect?
```

Arguments CONSte creates a new OFDM Constellation view.

SUMMary creates a new OFDM Summary view.

STABle creates a new OFDM Symbol table view.

CRESponse creates a new OFDM Channel response view.

EVM creates a new EVM (Error Vector Magnitude) versus Time view.

PERRor creates a new Phase error versus Time view.

MERRor creates a new Magnitude error versus Time view.

POWer creates a new OFDM Power view.

If you attempt to open a view that is currently displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Examples DISPLAY:OFDM:MEASVIEW:SELECT:STABle selects the OFDM symbol table view.

DISPlay:OFDM:MERRor:FREQuency[:SCALe]

Sets or queries the horizontal range of the OFDM Magnitude error graph.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:MERRor:FREQuency[:SCALe] <value>
DISPlay:OFDM:MERRor:FREQuency[:SCALe]?
```

Arguments <value>::= <NRf> specifies the horizontal range.

Examples `DISPLAY:OFDM:MERROR:FREQUENCY[:SCALE] 10E+6` sets the horizontal range to 10 MHz.

DISPlay:OFDM:MERRor:FREQuency[:SCALe]:AUTO (No Query Form)

Rescales the frequency automatically to fit the waveform to the screen in the OFDM Magnitude error graph.

Conditions Measurement views: OFDM

Group Display commands

Syntax `DISPlay:OFDM:MERRor:FREQuency[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:OFDM:MERROR:FREQUENCY[:SCALE]:AUTO` rescales the frequency scale automatically to fit the waveform to the screen.

DISPlay:OFDM:MERRor:FREQuency[:SCALe]:OFFSet

Sets or queries the frequency offset in the OFDM Magnitude error graph.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:MERRor:FREQuency[:SCALe]:OFFSet <value>
DISPlay:OFDM:MERRor:FREQuency[:SCALe]:OFFSet?
```

Arguments <value> :: <NRf> specifies the frequency offset.

Examples `DISPLAY:OFDM:MERROR:FREQUENCY[:SCALE]:OFFSET -26` sets the offset to –26 subcarriers.

DISPlay:OFDM:MERRor:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the OFDM Magnitude error view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:MERRor:MARKer:SHOW:STATe { OFF | ON | 1| 0 }
DISPlay:OFDM:MERRor:MARKer:SHOW:STATe?
```

Arguments `OFF` or `0` hides the readout for the selected marker in the graph.

`ON` or `1` shows the readout for the selected marker in the graph.

Examples `DISPLAY:OFDM:MERROR:MARKER:SHOW:STATE ON` shows the readout for the selected marker in the graph.

DISPlay:OFDM:MERRor:TIME[:SCALe]

Sets or queries the time scale in the OFDM Magnitude error view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:MERRor:TIME[:SCALe] <value>
DISPlay:OFDM:MERRor:TIME[:SCALe]?
```

Arguments <value>::= <NRf> specifies the time scale.

Examples `DISPLAY:OFDM:MERROR:TIME[:SCALE] 50` sets the time scale to 50 subcariers.

DISPlay:OFDM:MERRor:TIME[:SCALe]:AUTO (No Query Form)

Rescales the time automatically to fit the waveform to the screen in the OFDM Magnitude error view.

Conditions Measurement views: OFDM

Group Display commands

Syntax `DISPlay:OFDM:MERRor:TIME[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:OFDM:MERROR:TIME[:SCALE]:AUTO` rescales the time scale automatically to fit the waveform to the screen.

DISPlay:OFDM:MERRor:TIME[:SCALe]:OFFSet

Sets or queries the time offset in the OFDM Magnitude error view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:MERRor:TIME[:SCALe]:OFFSet <value>
DISPlay:OFDM:MERRor:TIME[:SCALe]:OFFSet?
```

Arguments <value>::= <NRf> specifies the time offset.

Examples `DISPLAY:OFDM:MERROR:TIME[:SCALE]:OFFSET -47` sets the offset to –47 symbols.

DISPlay:OFDM:MERRor:WINDow:SELect:PLOT

Sets or queries the plot in the OFDM Magnitude error window measurement.

Conditions Measurement views: OFDM

Group Display commands

Syntax

```
DISPlay:OFDM:MERRor:WINDow:SELect:PLOT { BOTH | SCARrier |
SYMBols }
DISPlay:OFDM:MERRor:WINDow:SELect:PLOT?
```

Arguments

`SCARrier` selects the subcarrier display.

`SYMBols` selects the symbols display

`BOTH` selects both displays.

Examples

`DISPLAY:OFDM:MERROR:WINDOW:SELECT:PLOT SYMBols` selects the symbols display.

DISPlay:OFDM:MERRor:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule in the OFDM Magnitude error view.

Conditions

Measurement views: OFDM

Group

Display commands

Syntax

```
DISPlay:OFDM:MERRor:WINDow:TRACe:GRATicule:GRID:STATe { OFF
| ON | 1| 0}
DISPlay:OFDM:MERRor:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or `0` hides the graticule in the graph.

`ON` or `1` shows the graticule in the graph.

Examples

`DISPLAY:OFDM:MERROR:WINDOW:TRACE:GRATICULE:GRID:STATE OFF` turns off the graticule in the graph.

DISPlay:OFDM:MERRor:Y:AUTO (No Query Form)

Rescales the vertical scale automatically to fit the waveform in the OFDM Magnitude error view.

Conditions

Measurement views: OFDM

Group Display commands

Syntax `DISPlay:OFDM:MERRor:Y:AUTO`

Arguments None

Examples `DISPLAY:OFDM:MERROR:Y:AUTO` rescales the vertical scale to fit the waveform on the screen.

DISPlay:OFDM:MERRor:Y:OFFSet

Sets or queries the vertical offset in the OFDM Magnitude error view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:MERRor:Y:OFFSet <value>
DISPlay:OFDM:MERRor:Y:OFFSet?
```

Arguments <value>::= <NRf> specifies the vertical offset.

Examples `DISPLAY:OFDM:MERROR:Y:OFFSET -90` sets the vertical scale offset to –90%.

DISPlay:OFDM:MERRor:Y[:SCALe]

Sets or queries the vertical range of the OFDM Magnitude error view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:MERRor:Y[:SCALe] <value>
DISPlay:OFDM:MERRor:Y[:SCALe]?
```

Arguments <value>::= <NRf> specifies the vertical scale.

Examples `DISPLAY:OFDM:MERROR:Y[:SCALE] 100` sets the vertical scale to 100%.

DISPlay:OFDM:PERRor:FREQuency[:SCALe]

Sets or queries the horizontal range of the OFDM Phase error graph.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:PERRor:FREQuency[:SCALe] <value>
DISPlay:OFDM:PERRor:FREQuency[:SCALe]?
```

Arguments <value>::= <NRf> specifies the horizontal range.

Examples `DISPLAY:OFDM:PERROR:FREQUENCY[:SCALE] 10E+6` sets the horizontal range to 10 MHz.

DISPlay:OFDM:PERRor:FREQuency[:SCALe]:AUTO (No Query Form)

Rescales the frequency automatically to fit the waveform to the screen in the OFDM Phase error graph.

Conditions Measurement views: OFDM

Group Display commands

Syntax `DISPlay:OFDM:PERRor:FREQuency[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:OFDM:PERROR:FREQUENCY[:SCALE]:AUTO` rescales the frequency scale automatically to fit the waveform to the screen.

DISPlay:OFDM:PERRor:FREQuency[:SCALe]:OFFSet

Sets or queries the frequency offset in the OFDM Phase error view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:PERRor:FREQuency[:SCALe]:OFFSet <value>
DISPlay:OFDM:PERRor:FREQuency[:SCALe]:OFFSet?
```

Arguments <value>::= <NRf> specifies the frequency offset.

Examples `DISPLAY:OFDM:PERROR:FREQUENCY[:SCALE]:OFFSET -26` sets the frequency scale offset to –26 subcarriers.

DISPlay:OFDM:PERRor:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the OFDM Phase error view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:PERRor:MARKer:SHOW:STATe { OFF | ON | 1| 0 }
DISPlay:OFDM:PERRor:MARKer:SHOW:STATe?
```

Arguments `OFF` or `0` hides the readout for the selected marker in the graph.

`ON` or `1` shows the readout for the selected marker in the graph.

Examples `DISPLAY:OFDM:PERROR:MARKER:SHOW:STATE ON` shows the readout for the selected marker in the graph.

DISPlay:OFDM:PERRor:TIME[:SCALe]

Sets or queries the time scale in the OFDM Phase error view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:PERRor:TIME[:SCALe] <value>
DISPlay:OFDM:PERRor:TIME[:SCALe]?
```

Arguments <value>::= <NRf> specifies the time scale.

Examples `DISPLAY:OFDM:PERROR:TIME[:SCALE] 50` sets the time scale to 50 symbols.

DISPlay:OFDM:PERRor:TIME[:SCALe]:AUTO (No Query Form)

Rescales the time automatically to fit the waveform to the screen in the OFDM Phase error view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:PERRor:TIME[:SCALe]:AUTO
```

Arguments None

Examples `DISPLAY:OFDM:PERROR:TIME[:SCALE]:AUTO` rescales the time scale automatically to fit the waveform to the screen.

DISPlay:OFDM:PERRor:TIME[:SCALe]:OFFSet

Sets or queries the time offset in the OFDM Phase error view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:PERRor:TIME[:SCALe]:OFFSet <value>
DISPlay:OFDM:PERRor:TIME[:SCALe]:OFFSet?
```

Arguments <value>::= <NRf> specifies the time offset.

Examples `DISPLAY:OFDM:PERROR:TIME[:SCALE]:OFFSET −17` sets the time scale offset to –17 symbols.

DISPlay:OFDM:PERRor:WINDow:SELect:PLOT

Sets or queries the plot in the OFDM Phase error window measurement.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:PERRor:WINDow:SELect:PLOT { BOTH | SCARrier |
SYMBols }
DISPlay:OFDM:PERRor:WINDow:SELect:PLOT?
```

Arguments `SCARrier` selects the subcarrier display.

`SYMBols` selects the symbols display

`BOTH` selects both displays.

Examples `DISPLAY:OFDM:PERROR:WINDOW:SELECT:PLOT SYMBols` selects the symbols display.

DISPlay:OFDM:PERRor:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule in the OFDM Phase error view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:PERRor:WINDow:TRACe:GRATicule:GRID:STATe { OFF
| ON | 1| 0}
DISPlay:OFDM:PERRor:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule in the graph.

`ON` or `1` shows the graticule in the graph.

Examples `DISPLAY:OFDM:PERROR:WINDOW:TRACE:GRATICULE:GRID:STATE OFF` turns off the graticule in the graph.

DISPlay:OFDM:PERRor:Y:AUTO (No Query Form)

Rescales the vertical scale automatically to fit the waveform in the OFDM Phase error view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:PERRor:Y:AUTO
```

Arguments None

Examples `DISPLAY:OFDM:PERROR:Y:AUTO` rescales the vertical scale to fit the waveform on the screen.

DISPlay:OFDM:PERRor:Y:OFFSet

Sets or queries the vertical offset in the OFDM Phase error view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:PERRor:Y:OFFSet <value>
DISPlay:OFDM:PERRor:Y:OFFSet?
```

Arguments <value>::= <NRf> specifies the horizontal range.

Examples `DISPLAY:OFDM:PERROR:Y:OFFSET` –173 sets the vertical offset to –173 °.

DISPlay:OFDM:PERRor:Y[:SCALe]

Sets or queries the vertical range of the OFDM Phase error view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:PERRor:Y[:SCALe] <value>
DISPlay:OFDM:PERRor:Y[:SCALe]?
```

Arguments <value>::= <NRf> specifies the vertical scale.

Examples `DISPLAY:OFDM:PERROR:Y[:SCALE] 200` sets the vertical scale to 200 °.

DISPlay:OFDM:POWer:FREQuency[:AUTO] (No Query Form)

Rescales the frequency automatically to fit the waveform to the screen in the OFDM Power graph.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:POWer:FREQuency[:AUTO]
```

Arguments None

Examples `DISPLAY:OFDM:POWER:FREQUENCY[:AUTO]` rescales the frequency scale automatically to fit the waveform to the screen.

DISPlay:OFDM:POWer:FREQuency[:OFFSet]

Sets or queries the frequency offset in the OFDM Power view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:POWer:FREQuency[:OFFSet] <value>
DISPlay:OFDM:POWer:FREQuency[:OFFSet]?
```

Arguments <value>::= <NRf> specifies the frequency offset.

Examples `DISPLAY:OFDM:POWER:FREQUENCY[:OFFSET] -26` sets the frequency offset to –26 subcarriers.

DISPlay:OFDM:POWer:FREQuency[:SCALe]

Sets or queries the horizontal range of the OFDM Power graph.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:POWer:FREQuency[:SCALe] <value>
DISPlay:OFDM:POWer:FREQuency[:SCALe]?
```

Arguments <value>::= <NRf> specifies the horizontal range.

Examples `DISPLAY:OFDM:POWER:FREQUENCY[:SCALE] 10E+6` sets the horizontal range to 10 MHz.

DISPlay:OFDM:POWer:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the OFDM Power view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:POWer:MARKer:SHOW:STATe { OFF | ON | 1| 0 }
```

Arguments `OFF` or `0` hides the readout for the selected marker in the graph.

`ON` or `1` shows the readout for the selected marker in the graph.

Examples `DISPLAY:OFDM:POWER:MARKER:SHOW:STATE ON` shows the readout for the selected marker in the graph.

DISPlay:OFDM:POWer:TIME[:AUTO (No Query Form)

Rescales the time automatically to fit the waveform to the screen in the OFDM Power view.

Conditions Measurement views: OFDM

Group Display commands

Syntax `DISPlay:OFDM:POWer:TIME[:AUTO`

Arguments None

Examples `DISPLAY:OFDM:POWER:TIME[:AUTO` rescales the time scale automatically to fit the waveform to the screen.

DISPlay:OFDM:POWer:TIME:OFFSet

Sets or queries the time offset in the OFDM Power view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:POWer:TIME:OFFSet <value>
DISPlay:OFDM:POWer:TIME:OFFSet?
```

Arguments <value>::= <NR1> specifies the time offset.

Examples DISPLAY:OFDM:POWER:TIME:OFFSET? might return 2.00000 indicating the offset is 2 symbols.

DISPlay:OFDM:POWer:TIME[:SCALe]

Sets or queries the time scale in the OFDM Phase error view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:POWer:TIME[:SCALe] <value>
DISPlay:OFDM:POWer:TIME[:SCALe]?
```

Arguments <value>::= <NRf> specifies the time scale.

Examples DISPLAY:OFDM:POWER:TIME[:SCALE] 46 sets the time scale to 46 symbols.

DISPlay:OFDM:POWer:WINDow:SELect:PLOT

Sets or queries the plot in the OFDM Power window measurement.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:POWer:WINDow:SELect:PLOT { BOTH | SCARrier |
SYMBols }
DISPlay:OFDM:POWer:WINDow:SELect:PLOT?
```

Arguments SCARier selects the subcarrier display.

SYMBols selects the symbols display

BOTH selects both displays.

Examples DISPLAY:OFDM:POWER:WINDOW:SELECT:PLOT SYMBols selects the symbols display.

DISPlay:OFDM:POWer:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule in the OFDM Power view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:POWer:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 1| 0}
DISPlay:OFDM:POWer:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule in the graph.

`ON` or `1` shows the graticule in the graph.

Examples `DISPLAY:OFDM:POWER:WINDOW:TRACE:GRATICULE:GRID:STATE OFF` turns off the graticule in the graph.

DISPlay:OFDM:POWer:Y:AUTO (No Query Form)

Rescales the vertical scale automatically to fit the waveform in the OFDM Power view.

Conditions Measurement views: OFDM

Group Display commands

Syntax `DISPlay:OFDM:POWer:Y:AUTO`

Arguments None

Examples `DISPLAY:OFDM:POWER:Y:AUTO` rescales the vertical scale to fit the waveform on the screen.

DISPlay:OFDM:POWer:Y:OFFSet

Sets or queries the vertical offset in the OFDM Power view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:POWer:Y:OFFSet <value>
DISPlay:OFDM:POWer:Y:OFFSet?
```

Arguments <value>::= <NRf> specifies the horizontal range.

Examples `DISPLAY:OFDM:POWER:Y:OFFSET –37.29` sets the vertical offset to –37.29 dBm.

DISPlay:OFDM:POWer:Y[:SCALe]

Sets or queries the vertical range of the OFDM Power view.

Conditions Measurement views: OFDM

Group Display commands

Syntax
```
DISPlay:OFDM:POWer:Y[:SCALe] <value>
DISPlay:OFDM:POWer:Y[:SCALe]?
```

Arguments <value>::= <NRf> specifies the vertical scale.

Examples `DISPLAY:OFDM:POWER:Y[:SCALE] 64.48` sets the vertical scale to 64.48 dB.

DISPlay:P25:CONSte:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries to show or hide the graticule grid on the screen in the P25 Constellation display.

Conditions Measurement view: P25 Constellation

Group Display commands

Syntax

```
DISPlay:P25:CONSte:WINDow:TRACe:GRATicule:GRID:STATe { OFF |
ON | 0 | 1 }
DISPlay:P25:CONSte:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Returns

`0` means that the graticule grid is hidden.

`1` means that the graticule grid is showing.

Examples

`DISPLAY:P25:CONSTE:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the screen.

DISPlay:P25:EDIagram:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen in the P25 Eye Diagram display.

Conditions

Measurement view: P25 Eye Diagram

Group

Display commands

Syntax

```
DISPlay:P25:EDIagram:WINDow:TRACe:GRATicule:GRID:STATe
{OFF|ON|0|1}
DISPlay:P25:EDIagram:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Returns

`0` means that the graticule grid is hidden.

`1` means that the graticule grid is showing.

Examples

`DISPlay:P25:EDIagram:WINDow:TRACe:GRATicule:GRID:STATe ON` shows the graticule grid on the screen in the P25 Eye Diagram display.

DISPlay:P25:EDIagram:Y[:SCALe]

Sets or queries the vertical scale (Hz) of the P25 Eye Diagram display.

Conditions Measurement view: P25 Eye Diagram

Group Display commands

Syntax

```
DISPlay:P25:EDIagram:Y[:SCALe] <NRf>
DISPlay:P25:EDIagram:Y[:SCALe]?
```

Arguments <value> :: <NRf> specifies the vertical scale value (no units).

Range is 1 to 100

Returns <NR3> = a floating point value with an exponent, which is the vertical scale value in Hz.

Examples `DISP:P25:EDI:Y:SCAL 0.0005` sets the vertical scale to 500 µHz in the display.

`DISP:P25:EDI:Y:SCAL ?` night return 500.0000000000E-6, indicating that the vertical scale is 500 µHz.

DISPlay:P25:EDIagram:Y[:SCALe]:AUTO (No Query Form)

Sets the vertical scale automatically to fit the waveform to the screen in the P25 Eye Diagram display.

Conditions Measurement view: P25 Eye Diagram

Group Display commands

Syntax

```
DISPlay:P25:EDIagram:Y[:SCALe]:AUTO
```

Arguments None.

Returns 0 indicates the vertical scale is set to automatically fit the waveform to the screen in the P25 Eye Diagram display.

1 indicates the vertical scale is not set to automatically fit the waveform to the screen in the P25 Eye Diagram display.

Examples `DISPLAY:P25:EDIAGRAM:Y[:SCALE]:AUTO` sets the vertical scale automatically to fit the waveform to the screen.

DISPlay:P25:EDIagram:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (center point of the vertical axis) in the P25 Eye Diagram display.

Conditions Measurement view: P25 Eye Diagram

Group Display commands

Syntax
```
DISPlay:P25:EDIagram:Y[:SCALe]:OFFSet
DISPlay:P25:EDIagram:Y[:SCALe]:OFFSet?
```

Arguments <value> :: <NRf> specifies the minimum vertical value. Range is -50 to +50 (no units).

Returns Vertical offset value.

Examples `DISPLAY:P25:EDIAGRAM:Y[:SCALE]:OFFSET` -0.5 sets the vertical offset to -0.5 in the P25 Eye Diagram display.

DISPlay:P25:MEASview:DELete (No Query Form)

Deletes the specified P25 measurement display.

Conditions Measurement view: P25 measurements

Group Display commands

Syntax `DISPlay:P25:MEASview:DELete { FDVT | EDI | PVT | CONS | SUMM }`

Related Commands DISPlay:P25:MEASview:NEW

DISPlay:P25:MEASview:SELect

Arguments

FDVT: Frequency Deviation vs Time display

EDI: P25 Eye Diagram display

PVT: Power vs Time display

CONS: P25 Constellation display

SUMM: P25 Summary display

Examples

`DISPLAY:P25:MEASVIEW:DELETE CONS` deletes the P25 Constellation display.

DISPlay:P25:MEASview:NEW (No Query Form)

Displays a new P25 measurement view.

Conditions

Measurement view: P25 measurements

Group

Display commands

Syntax

`DISPlay:P25:MEASview:NEW { FDVT | EDI | PVT | CONS | SUMM }`

Related Commands

DISPlay:P25:MEASview:SELect

DISPlay:P25:MEASview:DELete

Arguments

FDVT: Frequency Deviation vs Time display

EDI: P25 Eye Diagram display

PVT: Power vs Time display

CONS: P25 Constellation display

SUMM: P25 Summary display

NOTE. *If you attempt to open a view that is currently displayed on the screen, the error -200, "Execution error; Measurement is already running" will be returned.*

Examples

`DISPLAY:P25:MEASVIEW:NEW PVT` creates a new view of the Power vs Time display.

DISPlay:P25:MEASview:SELect

Selects a P25 measurement display. The query form returns the currently selected display.

Conditions Measurement view: P25 measurements

Group Display commands

Syntax `DISPlay:P25:MEASview:SELect { FDVT | EDI | PVT | CONS | SUMM }`

Related Commands DISPlay:P25:MEASview:NEW

DISPlay:P25:MEASview:DELete

Arguments FDVT: Frequency Deviation vs Time display

EDI: P25 Eye Diagram display

PVT: Power vs Time display

CONS: P25 Constellation display

SUMM: P25 Summary display

NOTE. *If you attempt to select a view that is not displayed on the screen, the error -200, "Execution error; Measurement not running" will be returned.*

Returns FDVT: Frequency Deviation vs Time display

EDI: P25 Eye Diagram display

PVT: Power vs Time display

CONS: P25 Constellation display

SUMM: P25 Summary display

Examples `DISPLAY:P25:MEASVIEW:SELECTCONS` selects the P25 Constellation display.

DISPlay:P25:PVTime:BURSt:X[:SCALe]

Sets or queries the value of the scale (width) value, in seconds, for the P25 Power vs. Time display when using horizontal Full Burst view.

Conditions Measurement view: P25 Power vs. Time

Group Display commands

Syntax
```
DISPlay:P25:PVTime:BURSt:X[:SCALe]
DISPlay:P25:PVTime:BURSt:X[:SCALe]?
```

Related Commands DISPlay:P25:PVTime:BURSt:X[:SCALe]:AUTO

Arguments Floating point number that represents the value of the scale (width), in seconds, when using horizontal Full Burst view.

Returns Horizontal scale value.

Examples `DISPLAY:P25:PVTIME:BURST:X[:SCALE] 5.0E-6` sets the width of the display to 5.000 µS for the horizontal Full Burst view.

DISPlay:P25:PVTime:BURSt:X[:SCALe]:AUTO (No Query Form)

Automatically sets the starting time (position) and scale (width) values for the best display in the horizontal Full Burst view.

Conditions Measurement view: P25 Power vs. Time

Group Display commands

Syntax
```
DISPlay:P25:PVTime:BURSt:X[:SCALe]:AUTO
```

Related Commands DISPlay:P25:PVTime:BURSt:X[:SCALe]

Examples `DISPLAY:P25:PVTIME:BURST:X[:SCALE]:AUTO` automatically sets the starting time (position) and scale (width) values for the best display.

DISPlay:P25:PVTime:BURSt:X[:SCALe]:OFFSet

Sets or queries the starting time (position) value, in seconds, for the P25 Power vs. Time display when using the horizontal Full Burst view.

Conditions Measurement view: P25 Power vs. Time

Group Display commands

Syntax
```
DISPlay:P25:PVTime:BURSt:X[:SCALe]:OFFSet
DISPlay:P25:PVTime:BURSt:X[:SCALe]:OFFSet?
```

Related Commands DISPlay:P25:PVTime:BURSt:X[:SCALe]:AUTO

Arguments Floating point number that represents the starting time (position) value, in seconds, when using horizontal Full Burst view.

Returns Horizontal offset value.

Examples `DISPLAY:P25:PVTIME:BURST:X[:SCALE]:OFFSET 5.0E-6` sets the starting time (position) of the graph to 5.000 µS for horizontal Full Burst view.

DISPlay:P25:PVTime:BURSt:X[:SCALe]:RESet (No Query Form)

Resets the starting time (position) and scale (width) values for the horizontal Full Burst view on the P25 Power vs. Time display.

Conditions Measurement view: P25 Power vs. Time

Group Display commands

Syntax
```
DISPlay:P25:PVTime:BURSt:X[:SCALe]:RESet
```

Related Commands DISPlay:P25:PVTime:BURSt:X[:SCALe]:AUTO

Examples `DISPLAY:P25:PVTIME:BURST:X[:SCALE]:RESET` resets the starting time (position) and scale (width) values for the horizontal Full Burst view.

DISPlay:P25:PVTime:FALL:X[:SCALe]

Sets or queries the scale (width) value, in seconds, for the P25 Power vs. Time display when using the horizontal Falling Edge view.

Conditions Measurement view: P25 Power vs. Time

Group Display commands

Syntax
```
DISPlay:P25:PVTime:FALL:X[:SCALe]<NRf>
DISPlay:P25:PVTime:FALL:X[:SCALe]?
```

Related Commands

Arguments Floating point number that represents the scale (width), in seconds, when using the horizontal Falling Edge view.

Returns Horizontal scale value.

Examples `DISPLAY:P25:PVTIME:FALL:X[:SCALE] 5.0E-6` sets the width of the graph to 5.000 µS when using the horizontal Falling Edge view.

DISPlay:P25:PVTime:FALL:X[:SCALe]:AUTO (No Query Form)

Automatically sets the starting time (position) and scale (width) values for the best display in the horizontal Falling Edge view.

Conditions Measurement view: P25 Power vs. Time

Group Display commands

Syntax
```
DISPlay:P25:PVTime:FALL:X[:SCALe]:AUTO
```

Related Commands DISPlay:P25:PVTime:FALL:X[:SCALe]:RESet

Examples `DISPLAY:P25:PVTIME:FALL:X[:SCALE]:AUTO` automatically sets the starting time (position) and scale (width) values for the best display.

DISPlay:P25:PVTime:FALL:X[:SCALe]:OFFSet

Sets or queries the value for the starting time (offset) of the graph, in seconds, for the P25 Power vs. Time display when using the horizontal Falling Edge view.

Conditions Measurement view: P25 Power vs. Time

Group Display commands

Syntax `DISPlay:P25:PVTime:FALL:X[:SCALe]:OFFSet`

Related Commands DISPlay:P25:PVTime:RISE:X[:SCALe]

Arguments Floating point number that represents the value of the starting time (offset) of the graph, in seconds, when using the horizontal Falling Edge view.

Returns Horizontal offset value.

Examples `DISPLAY:P25:PVTIME:FALL:X[:SCALE]:OFFSET 5.0` sets the starting time (position) of the graph to 5.000 µS when using the horizontal Falling Edge view.

DISPlay:P25:PVTime:FALL:X[:SCALe]:RESet (No Query Form)

Resets the starting time (position) and scale (width) values for the horizontal Falling Edge view on the P25 Power vs. Time display.

Conditions Measurement view: P25 Power vs. Time

Group Display commands

Syntax `DISPlay:P25:PVTime:FALL:X[:SCALe]:RESet`

Examples `DISPLAY:P25:PVTIME:FALL:X[:SCALE]:RESET` resets the starting time (position) and scale (width) values for the horizontal Falling Edge view.

DISPlay:P25:PVTime:MARKer:SHOW:STATe

Shows or hides the marker readout on the P25 Power vs. Time display.

Conditions Measurement view: P25 Power vs. Time

Group Display commands

Syntax `DISPlay:P25:PVTime:MARKer:SHOW:STATe`

Arguments ON or 1 specifies to show the marker readout. OFF or 0 specifies to turn the marker readout off.

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where:

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `DISPLAY:P25:PVTIME:MARKER:SHOW:STATE?` might return #43204 (3204-byte data), which represents the vertical values (power) for the P25 Power vs. Time trace.

DISPlay:P25:PVTime:RISE:X[:SCALe]

Sets or queries the scale (width) value, in seconds, for the P25 Power vs. Time display when using the horizontal Rising Edge view.

Conditions Measurement view: P25 Power vs. Time

Group Display commands

Syntax
```
DISPlay:P25:PVTime:RISE:X[:SCALe]
DISPlay:P25:PVTime:RISE:X[:SCALe]?
```

Arguments Floating point number that represents the scale (width) value, in seconds, for the horizontal Rising Edge view.

Returns Scale value.

Examples `DISPLAY:P25:PVTIME:RISE:X[:SCALE] 5.0E-6` sets the width of the graph to 5.000 µS for the horizontal Rising Edge view.

DISPlay:P25:PVTime:RISE:X[:SCALe]:AUTO (No Query Form)

Sets the scale (width) value, in seconds, for the P25 Power vs. Time display to automatic.

Conditions Measurement view: P25 Power vs. Time

Group Display commands

Syntax `DISPlay:P25:PVTime:RISE:X[:SCALe]:AUTO`

Related Commands DISPlay:P25:PVTime:RISE:X[:SCALe]:RESet

Examples `DISPLAY:P25:PVTIME:RISE:X[:SCALE]:AUTO` automatically sets the starting time (position) and scale (width) values for the best display.

DISPlay:P25:PVTime:RISE:X[:SCALe]:OFFSet

Sets or queries the starting time value (offset), in seconds, for the P25 Power vs. Time display when using the horizontal Rising Edge view.

Conditions Measurement view: P25 Power vs. Time

Group Display commands

Syntax `DISPlay:P25:PVTime:RISE:X[:SCALe]:OFFSet`

Related Commands DISPlay:P25:PVTime:RISE:X[:SCALe]

Arguments Floating point number that represents the value of the starting time (offset) of the graph, in seconds, for the horizontal Rising Edge view.

Returns Horizontal scale view

Examples `DISPLAY:P25:PVTIME:RISE:X[:SCALE]:OFFSET 5.0e-6` sets the starting time (position) of the graph to 5.000 µS for the horizontal Rising Edge view.

DISPlay:P25:PVTime:RISE:X[:SCALe]:RESet (No Query Form)

Resets the starting time (position) and scale (width) values for the horizontal Rising Edge view on the P25 Power vs. Time display.

Conditions Measurement view: P25 Power vs. Time

Group Display commands

Syntax

```
DISPlay:P25:PVTime:RISE:X[:SCALe]:RESet
```

Related Commands DISPlay:P25:PVTime:RISE:X[:SCALe]:OFFSet

Examples `DISPLAY:P25:PVTIME:RISE:X[:SCALE]:RESET` resets the starting time (position) and scale (width) values for the horizontal Rising Edge view.

DISPlay:P25:PVTime:WINDow:SELect:PLOT

Sets or queries which view to use for the P25 Power vs. Time display. Select from Full Burst, Rising Edge or Falling Edge.

Conditions Measurement view: P25 Power vs. Time

Group Display commands

Syntax

```
DISPlay:P25:PVTime:WINDow:SELect:PLOT{BURS|RISE|FALL}
DISPlay:P25:PVTime:WINDow:SELect:PLOT?
```

Related Commands

Arguments

- BURSt sets the view to Full Burst, which displays the entire packet, with vertical lines indicating the length of the burst.
- RISE sets the view to Rising Edge, which zooms the display into the interval around the burst rising edge, with vertical lines indicating the 10% to 90% Power-On Ramp time.
- FALL sets the view to Falling Edge, which zooms the display into the interval around the burst Falling Edge, with vertical lines indicating the 90% to 10% Power-Down Ramp time.

Returns `DISPlay:P25:PVTime:WINDow:SELect:PLOT BURS` sets the view to Full Burst.

Examples `DISPLAY:P25:PVTIME:WINDOW:SELECT:PLOT`? might return RISE, indicating that the view has been set to Rising Edge.

DISPlay:P25:PVTime:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries the graticule state in the Power vs Time display to on (showing) or off (hidden).

Conditions Measurement view: P25 Power vs Time

Group Display commands

Syntax
```
DISPlay:P25:PVTime:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:P25:PVTime:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule in the Power vs. Time display.

`ON` or `1` shows the graticule in the Power vs. Time display.

Returns `0` means the graticule in the Noise Temperature display is hidden (off).

`1` means the graticule in the Noise Temperature display is showing (on).

Examples `DISPLAY:P25:PVTIME:WINDOW:TRACE:GRATICULE:GRID:STATE ON` will turn on the graticule in the Noise Temperature display.

`DISPLAY:P25:PVTIME:WINDOW:TRACE:GRATICULE:GRID:STATE?` might return 1, indicating that the graticule in the Noise Temperature display is showing (on).

DISPlay:P25:PVTime:Y[:SCALe]

Sets or queries the vertical scale value for the P25 Power vs. Time display.

Conditions Measurement view: P25 Power vs. Time

Group Display commands

Syntax
```
DISPlay:P25:PVTime:Y[:SCALe] <NRf>
DISPlay:P25:PVTime:Y[:SCALe]?
```

Related Commands DISPlay:P25:PVTime:Y[:SCALe]:PDIVision

Arguments <NRf> = a numeric value for the vertical scale in dB.

Returns <NR2> = a floating point number without an exponent, which is the vertical scale value.

Examples `DISP:P25:PVT:Y:SCAL 10.5` sets the vertical scale value to 10.50 dB.

`DISP:P25:PVT:Y:SCAL ?` might return 10, indicating that the vertical scale is 10.00 dB.

DISPlay:P25:PVTime:Y[:SCALe]:AUTO (No Query Form)

Automatically selects the vertical scale and position values for the P25 Power vs Time display.

Conditions Measurement view: P25 Power vs Time

Group Display commands

Syntax
```
DISPlay:P25:PVTime:Y[:SCALe]:AUTO
```

Related Commands DISPlay:P25:PVTime:Y[:SCALe]:AUTO

Examples `DISP:P25:PVT:Y:SCAL:AUTO` specifies to reset the vertical scale and position (offset) values for the P25 Power vs. Time display.

DISPlay:P25:PVTime:Y[:SCALe]:OFFSet

Sets or queries the value of the vertical offset (top edge of the vertical axis) for the P25 Power vs. Time display.

Conditions Measurement view: P25 Power vs. Time

Group Display commands

Syntax
```
DISPlay:P25:PVTime:Y[:SCALe]:OFFSet<NRf>
DISPlay:P25:PVTime:Y[:SCALe]:OFFSet?
```

Related Commands DISPlay:P25:PVTime:Y[:SCALe]

Arguments Floating point number between limits that specifies the value of the vertical offset. Positive inputs move the plot down and negative inputs move the plot up (as viewed by the user).

Returns The value of the vertical offset for the P25 Power vs. Time display as a floating point number.

Examples `DISPLAY:P25:PVTIME:Y[:SCALE]:OFFSET` –10 sets the vertical offset to –10 dBm.

DISPlay:P25:PVTime:Y[:SCALe]:PDIVision

Sets or queries the value of the vertical scale for the P25 Power vs. Time display, in dB/division. This is only a visual control for panning the graph.

Conditions Measurement view: P25 Power vs. Time

Group Display commands

Syntax
```
DISPlay:P25:PVTime:Y[:SCALe]:PDIVision<NRf>
DISPlay:P25:PVTime:Y[:SCALe]:PDIVision?
```

Related Commands DISPlay:P25:PVTime:Y[:SCALe]

Arguments Floating point value that represents the value of the vertical scale, in dB/division, for the P25 Power vs. Time display.

Examples `DISPLAY:P25:PVTIME:Y[:SCALE]:PDIVISION` 20 sets the vertical scale to 20.0 dB/division.

DISPlay:P25:RADix

Set or queries the symbol radix for the P25 Symbol Table display.

Conditions Measurement view: P25 Symbol Table

Group Display commands

Syntax
```
DISPlay:P25:RADix { QUAT | BIN | MODS }
```

Arguments QUATernary: sets the symbols base to quaternary.

BINary: sets the symbols base to binary.

MODsymbols: sets the symbols to modular.

Returns QUAT: means that the symbols base is set to Quaternary.

BINary: means that the symbols base is set to Binary.

MODsymbols: means that the symbols base is set to ModSymbols (modular).

Examples
```
DISPLAY:P25:RADIX
```

DISPlay:PERRor:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries the graticule grid view state for the Phase error versus Time view.

Conditions Measurement views: Magnitude error versus Time

Group Display commands

Syntax
```
DISPlay:PERRor:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON
| 0 | 1 }
DISPlay:PERRor:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples `DISPLAY:PERROR:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the Phase error versus Time view.

DISPlay:PERRor:Y[:SCALe]

Sets or queries the vertical range of the Phase error versus Time graph.

Conditions Measurement views: Phase error versus Time

Group Display commands

Syntax
```
DISPlay:PERRor:Y[:SCALe] <value>
DISPlay:PERRor:Y[:SCALe]?
```

Related Commands DISPlay:PERRor:Y[:SCALe]:OFFSet

Arguments `<value> ::= <NRf>` specifies the vertical range. Range: 1 to 360°.

Examples `DISPLAY:PERROR:Y:SCALE 30` sets the vertical range to 30 ° in the Phase error versus Time graph.

DISPlay:PERRor:Y[:SCALe]:AUTO (No Query Form)

Sets the vertical scale automatically to fit the waveform to the screen in the Phase error versus Time view.

Conditions Measurement views: Phase error versus Time

Group Display commands

Syntax `DISPlay:PERRor:Y[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:PERROR:Y:SCALE:AUTO` sets the vertical scale automatically to fit the waveform to the screen.

DISPlay:PERRor:Y[:SCALe]:OFFSet

Sets or queries the minimum vertical value (bottom edge) of the Phase error versus Time graph.

Conditions Measurement views: Phase error versus Time

Group Display commands

Syntax
```
DISPlay:PERRor:Y[:SCALe]:OFFSet <value>
DISPlay:PERRor:Y[:SCALe]:OFFSet?
```

Related Commands DISPlay:PERRor:Y[:SCALe]

Arguments `<value> ::` `<NRf>` specifies the minimum vertical value. Range: -360 to +360°.

Examples `DISPLAY:PERROR:Y:SCALE:OFFSET -14.5` sets the minimum vertical value to -14.5 ° in the Phase Error versus Time graph.

DISPlay:PHVTime:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions Measurement views: Phase versus Time

Group Display commands

Syntax
```
DISPlay:PHVTime:WINDow:TRACe:GRATicule:GRID:STATe { OFF |
ON | 0 | 1 }
DISPlay:PHVTime:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples `DISPLAY:PHVTIME:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the Frequency versus Time view.

DISPlay:PHVTime:X[:SCALe]

Sets or queries the horizontal scale (full-scale time) of the Phase versus Time graph. Programming a specified scale sets DISPlay:PHVTime:X[:SCALe] AUTO:STATe OFF.

Conditions Measurement views: Phase versus Time

Group Display commands

Syntax

```
DISPlay:PHVTime:X[:SCALe] <value>
DISPlay:PHVTime:X[:SCALe]?
```

Related Commands DISPlay:PHVTime:X[:SCALe]:AUTO:STATe, DISPlay:PHVTime:X[:SCALe]:OFFSet

Arguments `<value>::={ <NRf> | MAXimum | MINimum }` specifies the horizontal scale in full-scale time. `MAXimum` and `MINimum` represent the upper and lower limits of the setting range, respectively.

Use the DISPlay:PHVTime:X[:SCALe]:MAXimum? and DISPlay:PHVTime:X[:SCALe]:OFFSet:MINimum? queries to get the upper and lower limit values of the setting range.

Examples `DISPLAY:PHVTIME:X:SCALE 1.5ms` sets the horizontal scale to 1.5 ms.

DISPlay:PHVTime:X[:SCALe]:AUTO (No Query Form)

Sets the horizontal scale automatically to fit the waveform to the screen in the Phase versus Time view. Executing this command sets DISPlay:PHVTime:X[SCALe]:AUTO:STATe ON.

Conditions Measurement views: Phase versus Time

Group Display commands

Syntax `DISPlay:PHVTime:X[:SCALe]:AUTO`

Related Commands DISPlay:PHVTime:X[:SCALe]:AUTO:STATe

Arguments None

Examples `DISPLAY:PHVTIME:X:SCALE:AUTO` sets the horizontal scale automatically to fit the waveform to the screen.

DISPlay:PHVTime:X[:SCALe]:AUTO:STATe

Determines whether to set the horizontal scale automatically or manually.

Conditions Measurement views: Phase versus Time

Group Display commands

Syntax
```
DISPlay:PHVTime:X[:SCALe]:AUTO:STATe { OFF | ON | 0 | 1 }
DISPlay:PHVTime:X[:SCALe]:AUTO:STATe?
```

Arguments `OFF` or `0` specifies that the horizontal scale is set manually. To set it, use the DISPlay:PHVTime:X[:SCALe] and DISPlay:PHVTime:X[:SCALe]:OFFSet commands.

`ON` or `1` specifies that the horizontal scale is set automatically.

Examples `DISPLAY:PHVTIME:X:SCALE:AUTO:STATE ON` specifies that the horizontal scale is set automatically.

DISPlay:PHVTime:X[:SCALe]:MAXimum? (Query Only)

Queries the upper limit of the horizontal scale setting range.

Conditions Measurement views: Phase versus Time

Group Display commands

Syntax `DISPlay:PHVTime:X[:SCALe]:MAXimum?`

Related Commands DISPlay:PHVTime:X[:SCALe]

Arguments None

Returns <NRf> The upper limit of the horizontal scale setting range.

Examples `DISPLAY:PHVTIME:X:SCALE:MAXIMUM?` might return `18.135E-3`, indicating that the upper limit of the horizontal scale setting range is 18.135 ms.

DISPlay:PHVTime:X[:SCALe]:MINimum? (Query Only)

Queries the lower limit of the horizontal scale setting range.

Conditions Measurement views: Phase versus Time

Group Display commands

Syntax `DISPlay:PHVTime:X[:SCALe]:MINimum?`

Related Commands DISPlay:PHVTime:X[:SCALe]

Arguments None

Returns <NRf> The lower limit of the horizontal scale setting range.

Examples `DISPLAY:PHVTIME:X:SCALE:MINIMUM?` might return `10.0E-9`, indicating that the lower limit of the horizontal scale setting range is 10.0 ns.

DISPlay:PHVTime:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (left edge) of the Phase versus Time graph. Programming a specified offset sets DISPlay:PHVTime:X [SCALe]:AUTO:STATe OFF.

Conditions Measurement views: Phase versus Time

Group Display commands

Syntax

```
DISPlay:PHVTime:X[:SCALe]:OFFSet
DISPlay:PHVTime:X[:SCALe]:OFFSet?
```

Related Commands DISPlay:PHVTime:X[:SCALe]:AUTO:STATe, DISPlay:PHVTime:X[:SCALe],

Arguments `<value>::={ <NRf> | MAXimum | MINimum }` specifies the horizontal offset. `MAXimum` and `MINimum` represent the upper and lower limits of the setting range, respectively.

Use the DISPlay:PHVTime:X[:SCALe]:OFFSet:MAXimum? and DISPlay:PHVTime:X[:SCALe]:OFFSet:MINimum? queries to get the upper and lower limit values of the setting range.

Examples `DISPLAY:PHVTIME:X:SCALE:OFFSET 800ns` sets the minimum horizontal value to 800 ns in the Phase versus Time graph.

DISPlay:PHVTime:X[:SCALe]:OFFSet:MAXimum? (Query Only)

Queries the upper limit of the horizontal offset setting range.

Conditions Measurement views: Phase versus Time

Group Display commands

Syntax `DISPlay:PHVTime:X[:SCALe]:OFFSet:MAXimum?`

Related Commands DISPlay:PHVTime:X[:SCALe]:OFFSet

Arguments None

Returns `<NRf>` The upper limit of the horizontal offset setting range.

Examples `DISPLAY:PHVTIME:X:SCALE:OFFSET:MAXIMUM?` might return `-1.812E-3`, indicating that the upper limit of the horizontal offset setting range is -1.812 ms.

DISPlay:PHVTime:X[:SCALe]:OFFSet:MINimum? (Query Only)

Queries the lower limit of the horizontal offset setting range.

Conditions Measurement views: Phase versus Time

Group Display commands

Syntax `DISPlay:PHVTime:X[:SCALe]:OFFSet:MINimum?`

Related Commands DISPlay:PHVTime:X[:SCALe]:OFFSet

Arguments None

Returns `<NRf>` The lower limit of the horizontal offset setting range.

Examples `DISPLAY:PHVTIME:X:SCALE:OFFSET:MINIMUM?` might return `-16.28E-3`, indicating that the lower limit of the horizontal offset setting range is -16.28 ms.

DISPlay:PHVTime:Y[:SCALe]

Sets or queries the vertical range of the Phase versus Time graph.

Conditions Measurement views: Phase versus Time

Group Display commands

Syntax
```
DISPlay:PHVTime:Y[:SCALe] <value>
DISPlay:PHVTime:Y[:SCALe]?
```

Related Commands DISPlay:PHVTime:Y[:SCALe]:OFFSet

Arguments `<value> ::` `<NRf>` specifies the vertical range. Range: 1 to 1T °.

Examples `DISPLAY:PHVTIME:Y:SCALE 180` sets the vertical range to 180 ° in the Phase versus Time graph.

DISPlay:PHVTime:Y[:SCALe]:AUTO (No Query Form)

Sets the vertical scale automatically to fit the waveform to the screen in the Phase versus Time view.

Conditions Measurement views: Phase versus Time

Group Display commands

Syntax `DISPlay:PHVTime:Y[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:PHVTIME:Y:SCALE:AUTO` sets the vertical scale automatically to fit the waveform to the screen.

DISPlay:PHVTime:Y[:SCALe]:AXIS

Sets or queries the vertical axis representation.

Conditions Measurement views: Phase versus Time

Group Display commands

Syntax
```
DISPlay:PHVTime:Y[:SCALe]:AXIS { MODulopi | CONTinuous }
DISPlay:PHVTime:Y[:SCALe]:AXIS?
```

Arguments `MODulopi` (modulo π) shows the phase constrained within ±180 ° along the vertical axis.

`CONTinuous` shows the phase as continuous quantity along the vertical axis.

Examples `DISPLAY:PHVTIME:Y:SCALE:AXIS MODulopi` selects modulo π representation for the vertical axis.

DISPlay:PHVTime:Y[:SCALe]:AXIS:REFerence

Sets or queries which time point in the analysis period to use as the zero-phase-value reference.

Conditions Measurement views: Phase versus Time

Group Display commands

Syntax
```
DISPlay:PHVTime:Y[:SCALe]:AXIS:REFerence <value>
DISPlay:PHVTime:Y[:SCALe]:AXIS:REFerence?
```

Arguments `<value> ::= <NRf>` specifies the phase reference time.

Examples `DISPLAY:PHVTIME:Y:SCALE:AXIS:REFERENCE 1.5us` sets the phase reference time to 1.5 µs.

DISPlay:PHVTime:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the center of the vertical axis) in the Phase versus Time graph.

Conditions Measurement views: Phase versus Time

Group Display commands

Syntax

```
DISPlay:PHVTime:Y[:SCALe]:OFFSet <value>
DISPlay:PHVTime:Y[:SCALe]:OFFSet?
```

Related Commands DISPlay:PHVTime:Y[:SCALe]

Arguments `<value> ::  <NRf>` specifies the vertical offset. Range: -0.5T to +0.5T°.

Examples `DISPLAY:PHVTIME:Y:SCALE:OFFSET -158.5` sets the vertical offset to -158.5 ° in the Phase versus Time graph.

DISPlay:PHVTime:Y[:SCALe]:RESCale (No Query Form)

Rescales the vertical axis automatically to fit the Phase versus Time waveform to the screen.

Conditions Measurement views: Phase versus Time

Group Display commands

Syntax

```
DISPlay:PHVTime:Y[:SCALe]:RESCale
```

Arguments None

Examples `DISPLAY:PHVTIME:Y:SCALE:RESCALE` rescales the vertical axis automatically to fit the Phase versus Time waveform to the screen.

DISPlay:PNOise:LEGend:STATe

Determines whether to show or hide the trace legend on the display. The legend indicates the trace detection and function on the screen for each displayed trace.

Conditions Measurement views: Phase versus Time

Group Display commands

Syntax
```
DISPlay:PNOise:LEGend:STATe { OFF | ON | 0 | 1 }
DISPlay:PNOise:LEGend:STATe?
```

Arguments `OFF` or `0` hides the trace legend.

`ON` or `1` shows the trace legend.

Examples `DISPLAY:PNOISE:LEGEND:STATE ON` shows the trace legend on the screen.

DISPlay:PNOise:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the phase noise view.

Conditions Measurement views: Phase noise

Group Display commands

Syntax
```
DISPlay:PNOise:MARKer:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:PNOise:MARKer:SHOW:STATe?
```

Arguments `OFF` or `0` hides the readout for the selected marker in the graph.

`ON` or `1` shows the readout for the selected marker in the graph.

Examples `DISPLAY:PNOISE:MARKER:SHOW:STATE ON` shows the readout for the selected marker in the graph.

DISPlay:PNOise:RESet:SCALe (No Query Form)

Resets the horizontal and vertical scale to the default values described below in the phase noise view.

Vertical offset = -50 dBc/Hz,
Vertical scale = 100 dB,
Horizontal start = 10 Hz, and
Horizontal stop = 1 GHz

Conditions Measurement views: Phase noise

Group Display commands

Syntax

```
DISPlay:PNOise:RESet:SCALe
```

Arguments None

Examples `DISPLAY:PNOISE:RESET:SCALE` resets the horizontal and vertical scale to the default values.

DISPlay:PNOise:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions Measurement views: Phase noise

Group Display commands

Syntax

```
DISPlay:PNOise:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:PNOise:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples `DISPLAY:PNOISE:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the screen.

DISPlay:PNOise:X[:SCALe]:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the phase noise view.

Conditions Measurement views: Phase noise

Group Display commands

Syntax `DISPlay:PNOise:X[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:PNOISE:X:SCALE:AUTO` rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:PNOise:X[:SCALe]:STARt

Sets or queries the start frequency (left edge) of the phase noise graph.

Conditions Measurement views: Phase noise

Group Display commands

Syntax
```
DISPlay:PNOise:X[:SCALe]:STARt <value>
DISPlay:PNOise:X[:SCALe]:STARt?
```

Arguments `<value> :: <NRf>` specifies the start frequency.
Range: 10 mHz to 100 MHz.

Note that (start frequency) = 10^4 × (stop frequency).

Examples `DISPLAY:PNOISE:X:SCALE:START 10Hz` sets the start frequency to 10 Hz in the phase noise graph.

DISPlay:PNOise:X[:SCALe]:STOP

Sets or queries the stop frequency (right edge) of the phase noise graph.

Conditions Measurement views: Phase noise

Group Display commands

Syntax
```
DISPlay:PNOise:X[:SCALe]:STOP <value>
DISPlay:PNOise:X[:SCALe]:STOP?
```

Arguments <value> :: <NRf> specifies the stop frequency. Range: 100 Hz to 1 THz.

Note that (start frequency) = $10^4 \times$ (stop frequency).

Examples DISPLAY:PNOISE:X:SCALE:STOP 2GHz sets the stop frequency to 2 GHz in the phase noise graph.

DISPlay:PNOise:Y[:SCALe]

Sets or queries the vertical range of the phase noise graph.

Conditions Measurement views: Phase noise

Group Display commands

Syntax
```
DISPlay:PNOise:Y[:SCALe] <value>
DISPlay:PNOise:Y[:SCALe]?
```

Arguments <value> :: <NRf> specifies the vertical range. Range: 0.1 to 200 dB.

Examples DISPLAY:PNOISE:Y:SCALE 100 sets the vertical range to 100 dB for the phase noise graph.

DISPlay:PNOise:Y[:SCALe]:AUTO (No Query Form)

Rescales the vertical axis automatically to fit the waveform to the screen in the phase noise view.

Conditions Measurement views: Phase noise

Group Display commands

Syntax `DISPlay:PNOise:Y[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:PNOISE:Y:SCALE:AUTO` rescales the vertical scale automatically to fit the waveform to the screen.

DISPlay:PNOise:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the top edge of the vertical axis) of the phase noise graph.

Conditions Measurement views: Phase noise

Group Display commands

Syntax
```
DISPlay:PNOise:Y[:SCALe]:OFFSet <value>
DISPlay:PNOise:Y[:SCALe]:OFFSet?
```

Arguments `<value> ::` `<NRf>` specifies the vertical offset. Range: -200 to +20 dBc/Hz.

Examples `DISPLAY:PNOISE:Y:SCALE:OFFSET -12.5` sets the vertical offset to -12.5 dBc/Hz for the phase noise graph.

DISPlay:PNOise:Y[:SCALe]:PDIVision

Sets or queries the vertical scale (per division) of the phase noise graph.

Conditions Measurement views: Phase noise

Group Display commands

Syntax

```
DISPlay:PNOise:Y[:SCALe]:PDIVision <value>
DISPlay:PNOise:Y[:SCALe]:PDIVision?
```

Arguments

<value> :: <NRf> specifies the vertical scale (per division).
Range: 0.01 to 20 dB/div.

Examples

DISPLAY:PNOISE:Y:SCALE:PDIVISION 5 sets the vertical scale to 5 dB/div.

DISPlay:PULSe:MEASview:DELete (No Query Form)

Deletes the measurement view in the pulsed RF measurements.

Conditions

Measurement views: Pulsed RF measurements

Group

Display commands

Syntax

```
DISPlay:PULSe:MEASview:DELete { RESult | TRACe | STATistics
}
```

Arguments

RESult deletes the pulse table view.

TRACe deletes the pulse trace view.

STATistics deletes the pulse statistics view.

If you attempt to delete a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running") will be returned.

Examples

DISPLAY:PULSE:MEASVIEW:DELETETRACe deletes the pulse trace view.

DISPlay:PULSe:MEASview:NEW (No Query Form)

Displays a new measurement view in the pulsed RF measurements.

Conditions

Measurement views: Pulsed RF measurements

Group

Display commands

Syntax

```
DISPlay:PULSe:MEASview:NEW { RESult | TRACe | STATistics }
```

Arguments

`RESult` opens the pulse table view.

`TRACe` opens the pulse trace view.

`STATistics` opens the pulse statistics view.

If you attempt to open a view that is currently displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Examples

`DISPLAY:PULSE:MEASVIEW:NEWSTATistics` creates the pulse statistics view.

DISPlay:PULSe:MEASview:SELect

Selects a measurement view in the pulsed RF measurements on the screen. The query command returns the currently selected view.

Conditions

Measurement views: Pulsed RF measurements

Group

Display commands

Syntax

```
DISPlay:PULSe:MEASview:SELect { RESult | TRACe | STATistics }
DISPlay:PULSe:MEASview:SELect?
```

Arguments

`RESult` selects the pulse table view.

`TRACe` selects the pulse trace view.

`STATistics` selects the pulse statistics view.

If you attempt to select a view that is not displayed on screen, the error (-200, "Execution error; Measurement not running") will be returned.

Examples

`DISPLAY:PULSE:MEASVIEW:SELECTTRACe` selects the pulse trace view.

DISPlay:PULSe:RESult:ATX

Determines whether or not to show the average transmitted power measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:ATX { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:ATX?
```

Arguments `OFF` or `0` does not show the average transmitted power measurement result.

`ON` or `1` shows the average transmitted power measurement result in the pulse table.

Examples `DISPLAY:PULSE:RESULT:ATX ON` shows the average transmitted power measurement result in the pulse table.

DISPlay:PULSe:RESult:AVERage

Determines whether or not to show the average on power measurement result in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:AVERage { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:AVERage?
```

Arguments `OFF` or `0` does not show the average on power measurement result.

`ON` or `1` shows the average on power measurement result in the results table.

Examples `DISPLAY:PULSE:RESULT:AVERAGE ON` shows the average on power measurement result in the pulse table.

DISPlay:PULSe:RESult:DRODb

Determines whether or not to show the droop measurement result in dB in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax

```
DISPlay:PULSe:RESult:DRODb { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:DRODb?
```

Arguments

`OFF` or `0` does not show the droop measurement result in dB.

`ON` or `1` shows the droop measurement result in dB in the pulse table.

Examples

`DISPLAY:PULSE:RESULT:DRODB ON` shows the droop measurement result in dB in the pulse table.

DISPlay:PULSe:RESult:DROop

Sets or queries showing the droop measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult:DROop { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:DROop?
```

Arguments

`OFF` or `0` does not show the droop measurement result.

`ON` or `1` shows the droop measurement result in the pulse table.

Examples

`DISPLAY:PULSE:RESULT:DROOP ON` shows the droop measurement result in the pulse table.

DISPlay:PULSe:RESult:DUTPct

Determines whether or not to show the duty factor (%) measurement result in the pulse table.

Conditions

Measurement views: Pulse table

Group

Display commands

Syntax

```
DISPlay:PULSe:RESult:DUTPct { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:DUTPct?
```

Arguments OFF or 0 does not show the duty factor measurement result.

ON or 1 shows the duty factor measurement result in the pulse table.

Examples DISPLAY:PULSE:RESULT:DUTPCT ON shows the duty factor (%) measurement result in the pulse table.

DISPlay:PULSe:RESult:DUTRatio

Determines whether or not to show the duty factor (ratio) measurement result in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:DUTRatio { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:DUTRatio?
```

Arguments OFF or 0 does not show the duty factor measurement result.

ON or 1 shows the duty factor measurement result in the pulse table.

Examples DISPLAY:PULSE:RESULT:DUTRATIO ON shows the duty factor (ratio) measurement result in the pulse table.

DISPlay:PULSe:RESult:FALL

Determines whether or not to show the fall time measurement result in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:FALL { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:FALL?
```

Arguments OFF or 0 does not show the fall time measurement result.

ON or 1 shows the fall time measurement result in the pulse table.

Examples DISPLAY:PULSE:RESULT:FALL ON shows the fall time measurement result in the pulse table.

DISPlay:PULSe:RESult:FDELta

Determines whether or not to show the delta frequency measurement result in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax DISPlay:PULSe:RESult:FDELta { OFF | ON | 0 | 1 }

Arguments OFF or 0 does not show the delta frequency measurement result.

ON or 1 shows the delta frequency measurement result in the pulse table.

Examples DISPLAY:PULSE:RESULT:FDELTA ON shows the delta frequency measurement result in the pulse table.

DISPlay:PULSe:RESult:FRDeviation

Determines whether or not to show the frequency deviation measurement result in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:FRDeviation { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:FRDeviation?
```

Arguments `OFF` or `0` does not show the frequency deviation measurement result.

`ON` or `1` shows the frequency deviation measurement result in the pulse table.

Examples `DISPLAY:PULSE:RESULT:FRDEVIATION ON` shows the frequency deviation measurement result in the pulse table.

DISPlay:PULSe:RESult:IRAMplitude

Sets or queries showing the impulse response amplitude measurement result in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:IRAMplitude { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:IRAMplitude?
```

Arguments `OFF` or `0` does not show the Impulse Response Amplitude measurement result.

`ON` or `1` shows the Impulse Response Amplitude measurement result in the results table.

Examples `DISPLAY:PULSE:RESULT:IRAMPLITUDE ON` shows the Impulse Response Amplitude measurement result in the pulse table.

DISPlay:PULSe:RESult:IRTime

Sets or queries showing the Impulse Response Time measurement result in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:IRTime { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:IRTime?
```

Arguments `OFF` or `0` does not show the Impulse Response Time measurement result.

`ON` or `1` shows the Impulse Response Time measurement result in the results table.

Examples `DISPLAY:PULSE:RESULT:IRTIME ON` shows the Impulse Response Time measurement result in the pulse table.

DISPlay:PULSe:RESult:MFReqerror

Determines whether or not to show the maximum frequency error measurement result in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:MFReqerror { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:MFReqerror?
```

Arguments `OFF` or `0` does not show the maximum frequency error measurement result.

`ON` or `1` shows the maximum frequency error measurement result in the pulse table.

Examples `DISPLAY:PULSE:RESULT:MFREQERROR ON` shows the maximum frequency error measurement result in the pulse table.

DISPlay:PULSe:RESult:MPHerror

Determines whether or not to show the maximum phase error measurement result in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:MPHerror { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:MPHerror?
```

Arguments `OFF` or `0` does not show the maximum phase error measurement result.

`ON` or `1` shows the maximum phase error measurement result in the pulse table.

Examples `DISPLAY:PULSE:RESULT:MPHERROR ON` shows the maximum phase error measurement result in the pulse table.

DISPlay:PULSe:RESult:OVEDb

Determines whether or not to show the Overshoot measurement result in dB in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:OVEDb { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:OVEDb?
```

Arguments `OFF` or `0` does not show the Overshoot measurement result in dB.

`ON` or `1` shows the Overshoot measurement result in dB in the pulse table.

Examples `DISPLAY:PULSE:RESULT:OVEDB ON` shows the Overshoot measurement result in dB in the pulse table.

DISPlay:PULSe:RESult:OVERshoot

Sets or queries whether or not to show the Overshoot measurement result in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:OVERshoot { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:OVERshoot?
```

Arguments OFF or 0 does not show the Overshoot measurement result.

ON or 1 shows the Overshoot measurement result in the pulse table.

Examples DISPLAY:PULSE:RESULT:OVERSHOOT ON shows the Overshoot measurement result in the pulse table.

DISPlay:PULSe:RESult:PHDeviation

Determines whether or not to show the phase deviation measurement result in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:PHDeviation { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:PHDeviation?
```

Arguments OFF or 0 does not show the phase deviation measurement result.

ON or 1 shows the phase deviation measurement result in the pulse table.

Examples DISPLAY:PULSE:RESULT:PHDEVIATION ON shows the phase deviation measurement result in the pulse table.

DISPlay:PULSe:RESult:PPFRequency

Determines whether or not to show the pulse-pulse carrier frequency measurement result in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:PPFRequency { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:PPFRequency?
```

Arguments OFF or 0 does not show the pulse-pulse carrier frequency measurement result.

ON or 1 shows the pulse-pulse carrier frequency measurement result in the pulse table.

Examples DISPLAY:PULSE:RESULT:PPFREQUENCY ON shows the pulse-pulse carrier frequency measurement result in the pulse table.

DISPlay:PULSe:RESult:PPOWer

Determines whether or not to show the peak power measurement result in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:PPOWer { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:PPOWer?
```

Arguments OFF or 0 does not show the peak power measurement result.

ON or 1 shows the peak power measurement result in the pulse table.

Examples DISPLAY:PULSE:RESULT:PPOWER ON shows the peak power measurement result in the pulse table.

DISPlay:PULSe:RESult:PPPHase

Determines whether or not to show the pulse-pulse carrier phase measurement result in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:PPPHase { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:PPPHase?
```

Arguments `OFF` or `0` does not show the pulse-pulse carrier phase measurement result.

`ON` or `1` shows the pulse-pulse carrier phase measurement result in the pulse table.

Examples `DISPLAY:PULSE:RESULT:PPPHASE ON` shows the pulse-pulse carrier phase measurement result in the pulse table.

DISPlay:PULSe:RESult:RINTerval

Determines whether or not to show the repetition interval measurement result in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:RINTerval { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:RINTerval?
```

Arguments `OFF` or `0` does not show the repetition interval measurement result.

`ON` or `1` shows the repetition interval measurement result in the results table.

Examples `DISPLAY:PULSE:RESULT:RINTERVAL ON` shows the repetition interval measurement result in the pulse table.

DISPlay:PULSe:RESult:RIPDb

Sets or queries showing the ripple measurement result in dB in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:RIPDb { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:RIPDb?
```

Arguments `OFF` or `0` does not show the ripple measurement result in dB.

`ON` or `1` shows the ripple measurement result in dB in the pulse table.

Examples `DISPLAY:PULSE:RESULT:RIPDB ON` shows the ripple measurement result in dB in the pulse table.

DISPlay:PULSe:RESult:RIPPle

Sets or queries showing the ripple measurement result in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:RIPPle { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:RIPPle?
```

Arguments `OFF` or `0` does not show the ripple measurement result.

`ON` or `1` shows the ripple measurement result in the pulse table.

Examples `DISPLAY:PULSE:RESULT:RIPPLE ON` shows the ripple measurement result in the pulse table.

DISPlay:PULSe:RESult:RISE

Determines whether or not to show the rise time measurement result in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:RISE { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:RISE?
```

Arguments `OFF` or `0` does not show the rise time measurement result.

`ON` or `1` shows the rise time measurement result in the pulse table.

Examples `DISPLAY:PULSE:RESULT:RISE ON` shows the rise time measurement result in the pulse table.

DISPlay:PULSe:RESult:RMSFreqerror

Determines whether or not to show the RMS frequency error measurement result in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:RMSFreqerror { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:RMSFreqerror?
```

Arguments `OFF` or `0` does not show the RMS frequency error measurement result.

`ON` or `1` shows the RMS frequency error measurement result in the pulse table.

Examples `DISPLAY:PULSE:RESULT:RMSFREQERROR ON` shows the RMS frequency error measurement result in the pulse table.

DISPlay:PULSe:RESult:RMSPherror

Determines whether or not to show the RMS phase error measurement result in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:RMSPherror { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:RMSPherror?
```

Arguments `OFF` or `0` does not show the RMS phase error measurement result.

`ON` or `1` shows the RMS phase error measurement result in the pulse table.

Examples `DISPLAY:PULSE:RESULT:RMSPHERROR ON` shows the RMS phase error measurement result in the pulse table.

DISPlay:PULSe:RESult:RRATe

Determines whether or not to show the repetition rate measurement result in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:RRATe { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:RRATe?
```

Arguments `OFF` or `0` does not show the repetition rate measurement result.

`ON` or `1` shows the repetition rate measurement result in the pulse table.

Examples `DISPLAY:PULSE:RESULT:RRATE ON` shows the repetition rate measurement result in the pulse table.

DISPlay:PULSe:RESult:TIME

Determines whether or not to show the time measurement result in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:TIME { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:TIME?
```

Arguments `OFF` or `0` does not show the time measurement result.

`ON` or `1` shows the time measurement result in the pulse table.

Examples `DISPLAY:PULSE:RESULT:TIME ON` shows the time measurement result in the pulse table.

DISPlay:PULSe:RESult:WIDTh

Determines whether or not to show the pulse width measurement result in the pulse table.

Conditions Measurement views: Pulse table

Group Display commands

Syntax
```
DISPlay:PULSe:RESult:WIDTh { OFF | ON | 0 | 1 }
DISPlay:PULSe:RESult:WIDTh?
```

Arguments `OFF` or `0` does not show the pulse width measurement result.

`ON` or `1` shows the pulse width measurement result in the pulse table.

Examples `DISPLAY:PULSE:RESULT:WIDTH ON` shows the pulse width measurement result in the pulse table.

DISPlay:PULSe:SELect:NUMBer

Sets or queries a pulse to measure. For the selected pulse, the statistics view indicates the measurement result while the table view highlights it, and the trace view displays the waveform.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Display commands

Syntax
```
DISPlay:PULSe:SELect:NUMBer <number>
DISPlay:PULSe:SELect:NUMBer?
```

Arguments `<number> ::  <NR1>` specifies the number of pulse to measure. Range: -(the number of acquired pulses before the time reference) to +(the number of acquired pulses after the time reference).

Zero (0) represents the pulse at the analysis time reference specified using the [SENSe]:ANALysis:REFerence command. The number of acquired pulses depends on the analysis range.

Examples `DISPLAY:WINDOW:SELECT:NUMBER -28` measures the pulse #-28.

DISPlay:PULSe:SELect:RESult

Sets or queries which result is shown in the pulse trace and statistics views.

Conditions Measurement views: Pulse statistics, Pulse trace

Group Display commands

Syntax

```
DISPlay:PULSe:SELect:RESult { AVERage | PPOWer | ATX |
WIDTh| RISE | FALL | RINTerval | RRATe | DUTPct | DUTRatio |
RIPPle| RIPDb | DROop | DRODb | OVERshoot | OVEDb | PPPHase
| PPFRequency | RMSFreqerror | MFReqerror| RMSPherror |
MPHerror | FRDeviation | FDELta | PHDeviation | IRAMplitude
| IRTime }
DISPlay:PULSe:SELect:RESult?
```

Arguments The following table lists the arguments.

Table 2-24: Pulse results

Argument	Result
AVERage	Average on power
PPOWer	Peak power
ATX	Average transmitted power
WIDTh	Pulse width
RISE	Rise time
FALL	Fall time
RINTerval	Repetition interval
RRATe	Repetition rate
DUTPct	Duty factor (%)
DUTRatio	Duty factor (ratio)
RIPPle	Ripple

Table 2-24: Pulse results (cont.)

Argument	Result
RIPDb	Ripple (dB)
DROop	Droop
DRODb	Droop (dB)
OVERshoot	Overshoot
OVEDb	Overshoot (dB)
PPPHase	Pulse-pulse carrier phase
PPFRequency	Pulse-pulse carrier frequency
RMSFreqerror	RMS frequency error
MFReqerror	Maximum frequency error
RMSPherror	RMS phase error
MPHerror	Maximum phase error
FRDeviation	Frequency deviation
PHDeviation	Phase deviation
FDELta	Delta frequency
IRAMplitude	Impulse Response Amplitude
IRTime	Impulse Response Sidelobe Offset Time from Mainlobe

Examples `DISPLAY:PULSE:SELECT:RESULTAVERage` shows the average on power result in the pulse trace and statistics views.

DISPlay:PULSe:STATistics:MARKer:SHOW:STATe

Determines whether to show or hide the marker readout in the statistics graph. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to FFT.

Conditions Measurement views: Pulse statistics

Group Display commands

Syntax

```
DISPlay:PULSe:STATistics:MARKer:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:PULSe:STATistics:MARKer:SHOW:STATe?
```

Arguments `OFF` or `0` hides the marker readout.

`ON` or `1` shows the marker readout.

Examples `DISPLAY:PULSE:STATISTICS:MARKER:SHOW:STATE ON` shows the marker readout in the statistics graph.

DISPlay:PULSe:STATistics:PLOT

Sets or queries how to show the statistics graph.

Conditions Measurement views: Pulse statistics

Group Display commands

Syntax
```
DISPlay:PULSe:STATistics:PLOT { TRENd | FFT | TTRend |
HISTogram }
DISPlay:PULSe:STATistics:PLOT?
```

Arguments `TRENd` shows the statistics result along with the pulse number.

`FFT` shows the statistics result transformed into the frequency domain by FFT.

`TTRend` shows the statistics result along with time.

`HISTogram` shows the histogram statistics results.

Examples `DISPLAY:PULSE:STATISTICS:PLOTTRENd` shows the statistics result along with the pulse number.

DISPlay:PULSe:STATistics:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid in the statistics view.

Conditions Measurement views: Pulse statistics

Group Display commands

Syntax
```
DISPlay:PULSe:STATistics:WINDow:TRACe:GRATicule:GRID:STATe {
OFF | ON | 0 | 1 }
DISPlay:PULSe:STATistics:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples `DISPLAY:PULSE:STATISTICS:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the statistics view.

DISPlay:PULSe:STATistics:X:RSCale (No Query Form)

Rescales the horizontal axis to fit the waveform to the screen in the statistics graph.

Conditions Measurement views: Pulse statistics

Group Display commands

Syntax `DISPlay:PULSe:STATistics:X:RSCale`

Arguments None

Examples `DISPLAY:PULSE:STATISTICS:X:RSCALE` rescales the horizontal axis of the statistics graph.

DISPlay:PULSe:STATistics:X[:SCALe]:NUMBer

Sets or queries the horizontal full scale in the statistics graph.

Conditions Measurement views: Pulse statistics

Group Display commands

Syntax
```
DISPlay:PULSe:STATistics:X[:SCALe]:NUMBer <value>
DISPlay:PULSe:STATistics:X[:SCALe]:NUMBer?
```

Related Commands DISPlay:PULSe:STATistics:X[:SCALe]:OFFSet

Arguments `<value> ::= <NRf>` specifies the horizontal full scale. The setting range depends on the DISPlay:PULSe:STATistics:PLOT command parameters as shown in the table below.

DISPlay:PULSe:STATistics:PLOT	Setting range
TRENd	1 to 1000
FFT	1 Hz to 120 MHz

Examples `DISPlay:PULSe:STATistics:X:SCALe:NUMBer 50` sets the horizontal full scale to 50 pulses when the plot is trend.

DISPlay:PULSe:STATistics:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (the first pulse to show) in the statistics graph.

Conditions Measurement views: Pulse statistics

Group Display commands

Syntax
```
DISPlay:PULSe:STATistics:X[:SCALe]:OFFSet <value>
DISPlay:PULSe:STATistics:X[:SCALe]:OFFSet?
```

Arguments `<value> ::= <NRf>` specifies the number of the first pulse.
Range: -(X - X/10) to +(X - X/10)
where X is the horizontal scale set by the DISPlay:PULSe:STATistics:X[:SCALe]:NUMBer command.

Examples `DISPLAY:PULSE:STATISTICS:X:SCALE:OFFSET 120` sets the first pulse number to #120.

DISPlay:PULSe:STATistics:Y:RSCale (No Query Form)

Rescales the vertical axis to fit the waveform to the screen in the statistics graph.

Conditions Measurement views: Pulse statistics

Group Display commands

Syntax `DISPlay:PULSe:STATistics:Y:RSCale`

Arguments None

Examples `DISPLAY:PULSE:STATISTICS:Y:RSCALE` rescales the vertical axis of the statistics graph.

DISPlay:PULSe:STATistics:Y[:SCALe]:FULL

Sets or queries the vertical full scale in the statistics graph.

Conditions Measurement views: Pulse statistics

Group Display commands

Syntax

```
DISPlay:PULSe:STATistics:Y[:SCALe]:FULL <value>
DISPlay:PULSe:STATistics:Y[:SCALe]:FULL?
```

Related Commands DISPlay:PULSe:STATistics:X[:SCALe]:OFFSet

Arguments `<value> ::= <NRf>` specifies the vertical full scale. The setting range depends on the DISPlay:PULSe:STATistics:PLOT command parameters and the measurement items as shown in the following table.

DISPlay:PULSe:STATistics:PLOT	Measurement item	Setting range
TRENd	Average on power, Peak power, Average transmitted power	0.1 to 200 dB
	Pulse width, Rise time, Fall time, Repetition interval	1 n to 5 Ms
	Repetition rate	100 m to 100 MHz
	Duty factor, Ripple	1 to 100%
	Droop	1 to 200%
	Pulse-pulse carrier phase	1 to 360°
FFT	All	10 to 200 dB

Examples `DISPLAY:PULSE:STATISTICS:Y:SCALE:FULL 100` sets the vertical full scale to 100 dB.

DISPlay:PULSe:STATistics:Y[:SCALe]:OFFSet

Sets or queries the vertical offset in the statistics graph.

Conditions Measurement views: Pulse statistics

Group Display commands

Syntax

```
DISPlay:PULSe:STATistics:Y[:SCALe]:OFFSet <value>
DISPlay:PULSe:STATistics:Y[:SCALe]:OFFSet?
```

Arguments <value> :: <NRf> specifies the vertical offset. The vertical offset is the value at the top or the bottom edge of the graph depending on measurement items. The setting range depends on the DISPlay:PULSe:STATistics:PLOT command parameters and the measurement items as shown in the table below.

DISPlay:PULSe :STATistics:PLOT	Measurement item	Setting range	Offset position [1]
TRENd	Average on power, Peak power, Average transmitted power	-170 to +50 dBm	Top (Bottom for the unit of Volts or Watts)
	Pulse width, Rise time, Fall time, Repetition interval	0 to 5 Ms	Bottom
	Repetition rate	0 to 100 MHz	Bottom
	Duty factor, Ripple	0 to 100%	Bottom
	Droop	0 to +100%	Bottom
	Pulse-pulse carrier phase	-180 to +180°	Bottom
FFT	All	-400 to +100 dB	Top

1 ndicates whether the offset is the value at the top or the bottom edge of the graph.

Examples DISPLAY:PULSE:STATISTICS:Y:SCALE:OFFSET 24.8 sets the maximum vertical value to 24.8 dBm in the statistics graph.

DISPlay:PULSe:STATistics:Y[:SCALe]:STOP? (Query Only)

Queries the minimum vertical value (bottom edge) in the statistics graph.

Conditions Measurement views: Pulse statistics

Group Display commands

Syntax

```
DISPlay:PULSe:STATistics:Y[:SCALe]:STOP?
```

Related Commands DISPlay:PULSe:STATistics:X[:SCALe]:OFFSet

Arguments None

Returns <y_stop> :: <NRf> is the minimum vertical value (bottom edge).

Examples DISPLAY:PULSE:STATISTICS:Y:SCALE:STOP? might return -150.0, indicating that the minimum vertical value is -150 dBm in the pulse statistics graph.

DISPlay:PULSe:TRACe:MARKer:SHOW:STATe

Determines whether to show or hide the marker readout in the pulse trace view.

Conditions Measurement views: Pulse trace

Group Display commands

Syntax
```
DISPlay:PULSe:TRACe:MARKer:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:PULSe:TRACe:MARKer:SHOW:STATe?
```

Arguments OFF or 0 hides the marker readout.

ON or 1 shows the marker readout.

Examples DISPLAY:PULSE:TRACE:MARKER:SHOW:STATE ON shows the marker readout on the pulse trace view.

DISPlay:PULSe:TRACe:POINt:SHOW

Determines whether to show or hide the measurement points and lines in the pulse trace view.

Conditions Measurement views: Pulse trace

Group Display commands

Syntax
```
DISPlay:PULSe:TRACe:POINt:SHOW { OFF | ON | 0 | 1 }
DISPlay:PULSe:TRACe:POINt:SHOW?
```

Arguments `OFF` or `0` hides the measurement points and lines.

`ON` or `1` shows the measurement points and lines.

Examples `DISPLAY:PULSE:TRACE:POINT:SHOW ON` shows the measurement points and lines in the pulse trace view.

DISPlay:PULSe:TRACe:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid in the pulse trace view.

Conditions Measurement views: Pulse trace

Group Display commands

Syntax
```
DISPlay:PULSe:TRACe:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:PULSe:TRACe:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples `DISPLAY:PULSE:TRACE:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the pulse trace view.

DISPlay:PULSe:TRACe:X:RSCale (No Query Form)

Rescales the horizontal axis to fit the waveform to the screen in the pulse trace view.

Conditions Measurement views: Pulse trace

Group Display commands

Syntax
```
DISPlay:PULSe:TRACe:X:RSCale
```

Arguments None

Examples `DISPLAY:PULSE:TRACE:X:RSCALE` rescales the horizontal axis in the pulse trace view.

DISPlay:PULSe:TRACe:X[:SCALe]

Sets or queries the horizontal full scale in the pulse trace view.

Conditions Measurement views: Pulse trace

Group Display commands

Syntax
```
DISPlay:PULSe:TRACe:X[:SCALe] <value>
DISPlay:PULSe:TRACe:X[:SCALe]?
```

Arguments `<value> ::= <NRf>` specifies the horizontal full scale.
Range: 10 ns to acquisition memory capacity.

Examples `DISPLAY:PULSE:TRACE:X:SCALE 5.5E-6` sets the horizontal scale to 5.5 µ s.

DISPlay:PULSe:TRACe:X[:SCALe]:FULL

Sets or queries the full-scale reference for the horizontal rescale.

Conditions Measurement views: Pulse trace

Group Display commands

Syntax
```
DISPlay:PULSe:TRACe:X[:SCALe]:FULL { SELected | MAXimum }
DISPlay:PULSe:TRACe:X[:SCALe]:FULL?
```

Arguments `SELected` uses the selected pulse for the full-scale reference.

`MAXimum` uses the maximum pulse for the full-scale reference.

Examples `DISPLAY:PULSE:TRACE:X:SCALE:FULL SELected` uses the selected pulse for the full-scale reference.

DISPlay:PULSe:TRACe:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (left edge) in the pulse trace view.

Conditions Measurement views: Pulse trace

Group Display commands

Syntax
```
DISPlay:PULSe:TRACe:X[:SCALe]:OFFSet <value>
DISPlay:PULSe:TRACe:X[:SCALe]:OFFSet?
```

Related Commands DISPlay:PULSe:TRACe:X[:SCALe]:PDIVision

Arguments `<value> ::= <NRf>` specifies the minimum horizontal value.
Range: [(analysis offset) - (X scale) × 0.9] to [(analysis offset) + (analysis length) - (X scale) × 0.1]

Examples `DISPLAY:PULSE:TRACE:X:SCALE:OFFSET 937.5E-9` sets the minimum horizontal value to 937.5 ns.

DISPlay:PULSe:TRACe:X[:SCALe]:PDIVision

Sets or queries the horizontal full scale in the pulse trace view.

Conditions Measurement views: Pulse trace

Group Display commands

Syntax
```
DISPlay:PULSe:TRACe:X[:SCALe]:PDIVision <value>
DISPlay:PULSe:TRACe:X[:SCALe]:PDIVision?
```

Arguments `<value> ::= <NRf>` specifies the horizontal full scale.
Range: 10 ns to acquisition memory capacity.

Examples `DISPLAY:PULSE:TRACE:X:SCALE:PDIVISION 5.5E-6` sets the horizontal scale to 5.5 µs.

DISPlay:PULSe:TRACe:Y:RSCale (No Query Form)

Rescales the vertical axis to fit the waveform to the screen in the pulse trace view.

Conditions Measurement views: Pulse trace

Group Display commands

Syntax `DISPlay:PULSe:TRACe:Y:RSCale`

Arguments None

Examples `DISPLAY:PULSE:TRACE:Y:RSCALE` rescales the vertical axis in the pulse trace view.

DISPlay:PULSe:TRACe:Y[:SCALe]:FULL

Sets or queries the vertical full scale in the pulse trace view.

Conditions Measurement views: Pulse trace

Group Display commands

Syntax
```
DISPlay:PULSe:TRACe:Y[:SCALe]:FULL <value>
DISPlay:PULSe:TRACe:Y[:SCALe]:FULL?
```

Related Commands DISPlay:PULSe:TRACe:Y[:SCALe]:OFFSet

Arguments `<value> :: <NRf>` specifies the vertical full scale.
Range: 0.1 to 200 dB.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `DISPLAY:PULSE:TRACE:Y:SCALE:FULL 100` sets the vertical full scale to 100 dB.

DISPlay:PULSe:TRACe:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the top edge of the vertical axis) in the pulse trace view.

Conditions Measurement views: Pulse trace

Group Display commands

Syntax
```
DISPlay:PULSe:TRACe:Y[:SCALe]:OFFSet <value>
DISPlay:PULSe:TRACe:Y[:SCALe]:OFFSet?
```

Related Commands DISPlay:PULSe:TRACe:Y[:SCALe]:STOP?

Arguments `<value> ::` `<NRf>` specifies the vertical offset. Range: -170 to +50 dBm.

Examples `DISPLAY:PULSE:TRACE:Y:SCALE:OFFSET 23.5` sets the vertical offset to 23.5 dBm.

DISPlay:PULSe:TRACe:Y[:SCALe]:STOP? (Query Only)

Queries the minimum vertical value (bottom edge) in the pulse trace view.

Conditions Measurement views: Pulse trace

Group Display commands

Syntax
```
DISPlay:PULSe:TRACe:Y[:SCALe]:STOP?
```

Related Commands DISPlay:PULSe:TRACe:Y[:SCALe]:OFFSet

Arguments None

Returns `<y_stop> ::` `<NRf>` is the minimum vertical value (bottom edge).

Examples `DISPLAY:PULSE:TRACE:Y:SCALE:STOP?` might return `-150.0`, indicating that the minimum vertical value is -150 dBm in the pulse trace view.

DISPlay:SEM:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the Spectral Emissions Mask view.

Conditions Measurement views: Spectral Emissions Mask

Group Display commands

Syntax

```
DISPlay:SEM:MARKer:SHOW:STATe { OFF | ON | 1| 0 }
DISPlay:SEM:MARKer:SHOW:STATe?
```

Arguments `OFF` or `0` hides the readout for the selected marker in the graph.

`ON` or `1` shows the readout for the selected marker in the graph.

Examples `DISPLAY:SEM:MARKER:SHOW:STATE ON` shows the readout for the selected marker in the graph.

DISPlay:SEM:RESet:SCALe (No Query Form)

Resets the scale in the Spectral Emissions Mask view.

Conditions Measurement views: Spectral Emissions Mask

Group Display commands

Syntax

```
DISPlay:SEM:RESet:SCALe
```

Arguments None

Examples `DISPLAY:SEM:RESET:SCALE` resets the scale in the view.

DISPlay:SEM:SHOW:LIMit

Sets or queries the appearance of the limits in the Spectral Emissions Mask view

Conditions Measurement views: Spectral Emissions Mask

Group Display commands

Syntax
```
DISPlay:SEM:SHOW:LIMit { SHADed | LINE | OFF }
DISPlay:SEM:SHOW:LIMit?
```

Arguments `SHADed` uses shading for the limit area in the display.

`LINE` uses a line for the limit area.

`OFF` turns off the limits in the display.

Examples `DISPLAY:SEM:SHOW:LIMIT SHADed` uses shading to identify the limits in the display.

DISPlay:SEM:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule in the Spectral Emissions Mask view.

Conditions Measurement views: Spectral Emissions Mask

Group Display commands

Syntax
```
DISPlay:SEM:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON |
1| 0}
DISPlay:SEM:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule in the graph.

`ON` or `1` shows the graticule in the graph.

Examples `DISPLAY:SEM:WINDOW:TRACE:GRATICULE:GRID:STATE OFF` turns off the graticule in the graph.

DISPlay:SEM:X[:SCALe]:AUTO (No Query Form)

Rescales the horizontal scale automatically to fit the waveform in the Spectral Emissions Mask view.

Conditions Measurement views: Spectral Emissions Mask

Group Display commands

Syntax `DISPlay:SEM:X[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:SEM:X[:SCALE]:AUTO` rescales the horizontal scale to fit the waveform on the screen.

DISPlay:SEM:X[:SCALe]:START

Sets or queries the start frequency (left edge) of the Spectral Emissions Mask graph.

Conditions Measurement views: Spectral Emissions Mask

Group Display commands

Syntax
```
DISPlay:SEM:X[:SCALe]:START <value>
DISPlay:SEM:X[:SCALe]:START?
```

Arguments <value> :: <NRf> specifies the start frequency.

Examples `DISPLAY:SEM:X[:SCALE]:START 10Hz` sets the start frequency to 10 Hz in the graph.

DISPlay:SEM:X[:SCALe]:STOP

Sets or queries the stop frequency (right edge) of the Spectral Emissions Mask graph.

Conditions Measurement views: Spectral Emissions Mask

Group Display commands

Syntax
```
DISPlay:SEM:X[:SCALe]:STOP <value>
DISPlay:SEM:X[:SCALe]:STOP?
```

Arguments <value> :: <NRf> specifies the start frequency.

Examples `DISPLAY:SEM:X[:SCALE]:STOP 2GHz` sets the stop frequency to 2 GHz in the graph.

DISPlay:SEM:Y[:SCALe]

Sets or queries the vertical range of the Spectral Emissions Mask view.

Conditions Measurement views: Spectral Emissions Mask

Group Display commands

Syntax
```
DISPlay:SEM:Y[:SCALe] <value>
DISPlay:SEM:Y[:SCALe]?
```

Arguments <value>::= <NRf> specifies the vertical scale.

Examples `DISPLAY:SEM:Y[:SCALE] 200` sets the vertical scale to 200 MHz.

DISPlay:SEM:Y[:SCALe]:AUTO (No Query Form)

Rescales the vertical scale automatically to fit the waveform in the Spectral Emissions Mask view.

Conditions Measurement views: Spectral Emissions Mask

Group Display commands

Syntax `DISPlay:SEM:Y[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:SEM:Y[:SCALE]:AUTO` rescales the vertical scale to fit the waveform on the screen.

DISPlay:SEM:Y[:SCALe]:OFFSet

Sets or queries the vertical offset in the Spectral Emissions Mask view.

Conditions Measurement views: Spectral Emissions Mask

Group Display commands

Syntax

```
DISPlay:SEM:Y[:SCALe]:OFFSet <value>
DISPlay:SEM:Y[:SCALe]:OFFSet?
```

Arguments <value>::= <NRf> specifies the horizontal range.

Examples `DISPLAY:SEM:Y[:SCALE]:OFFSET 40` sets the vertical position to 40 dBm.

DISPlay:SGRam:FREQuency:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the spectrogram view.

Conditions Measurement views: Spectrogram

Group Display commands

Syntax

```
DISPlay:SGRam:FREQuency:AUTO
```

Arguments None

Examples `DISPLAY:SGRAM:FREQUENCY:AUTO` rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:SGRam:FREQuency:OFFSet

Sets or queries the frequency offset (the value at the center of the horizontal axis) in the spectrogram.

Conditions Measurement views: Spectrogram

Group Display commands

Syntax
```
DISPlay:SGRam:FREQuency:OFFSet <value>
DISPlay:SGRam:FREQuency:OFFSet?
```

Related Commands DISPlay:SPECtrum:FREQuency[:SCAle]

Arguments <value> :: <NRf> specifies the frequency offset.
Range: [(center frequency) - (X scale) × 0.9] to [(center frequency) + (X scale) × 0.9]

Examples DISPLAY:SGRAM:FREQUENCY:OFFSET 1.45GHz sets the frequency offset to 1.45 GHz in the spectrogram.

DISPlay:SGRam:FREQuency:SCAle

Sets or queries the horizontal range of the spectrogram.

Conditions Measurement views: Spectrogram

Group Display commands

Syntax
```
DISPlay:SGRam:FREQuency:SCAle <value>
DISPlay:SGRam:FREQuency:SCAle?
```

Related Commands DISPlay:SPECtrum:FREQuency:OFFSet

Arguments <value> :: <NRf> specifies the horizontal range.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples DISPLAY:SGRAM:FREQUENCY:SCALE 10MHz sets the horizontal range to 10 MHz.

DISPlay:SGRAM:MARKer:SHOW:STATe

Sets or queries the display of the Marker legend in the Spectrogram graph.

Conditions Measurement views: Spectrogram

Group Display commands

Syntax

```
DISPlay:SGRAM:MARKer:SHOW:STATe
DISPlay:SGRAM:MARKer:SHOW:STATe?
```

Arguments OFF or 0 removes or turns off the Marker legend.

ON or 1 displays or turns on the Marker legend.

Examples DISPLAY:SGRAM:MARKER:SHOW:STATE ON turns on the display of the Marker legend in the spectrogram graph.

DISPlay:SGRAM:SELected:TIMestamp

Sets or queries the display of the time stamp in the Spectrogram graph.

Conditions Measurement views: Spectrogram

Group Display commands

Syntax

```
DISPlay:SGRAM:SELected:TIMestamp
DISPlay:SGRAM:SELected:TIMestamp?
```

Arguments OFF or 0 removes or turns off the time stamp.

ON or 1 displays or turns on the time stamp.

Examples DISPLAY:SGRAM:SELECTED:TIMESTAMP? queries the display status of the time stamp in the spectrogram graph.

DISPlay:SGRam:TIME:AUTO (No Query Form)

Rescales the vertical axis automatically to fit the graph to the screen in the spectrogram view.

Conditions Measurement views: Spectrogram

Group Display commands

Syntax
```
DISPlay:SGRam:TIME:AUTO
```

Arguments None

Examples `DISPLAY:SGRAM:TIME:AUTO` rescales the vertical scale automatically to fit the waveform to the screen.

DISPlay:SGRam:TIME:OFFSet

Sets or queries the vertical axis (time) offset (bottom line number) in the spectrogram.

Conditions Measurement views: Spectrogram

Group Display commands

Syntax
```
DISPlay:SGRam:TIME:OFFSet <value>
DISPlay:SGRam:TIME:OFFSet?
```

Related Commands DISPlay:SGRam:TIME:SCAle

Arguments `<value> ::  <NRf>` specifies the time offset.
Range: Line #0 to 125000. Zero (0) represents the latest line.

Examples `DISPLAY:SGRAM:TIME:OFFSET 15` sets the time offset to Line #15.

DISPlay:SGRam:TIME:OFFSet:DIVisions

Sets or queries the Vertical offset or Position in divisions.

Conditions Measurement views: Spectrogram

Group Display commands

Syntax
```
DISPlay:SGRam:TIME:OFFSet:DIVisions <value>
DISPlay:SGRam:TIME:OFFSet:DIVisions?
```

Arguments `<value> ::= <NRf>` sets the vertical offset value in divisions. Range is -130,000 to +130,000.

Examples `DISPLAY:SGRAM:TIME:OFFSET:DIVISIONS 10` sets the vertical offset to 10 divisions

DISPlay:SGRam:TIME:OVERlap:PERCent

Sets or queries the Overlap in percent. Note that changing this value may change the time per divisionDISPlay:SGRam:TIME:SCALe:PER:DIVision value.

Conditions Measurement views: Spectrogram

Group Display commands

Syntax
```
DISPlay:SGRam:TIME:OVERlap:PERCent
DISPlay:SGRam:TIME:OVERlap:PERCent
```

Arguments `<value> ::= <NRf>` specifies the overlap in percent.

Examples `DISPLAY:SGRAM:TIME:OVERLAP:PERCENT 1.0E1` sets the overlap to 10%.

DISPlay:SGRam:TIME:SCAle

NOTE. *The user interface on the instrument no longer presents this setting. However, the command is still function. Setting this value will change the TIME:PER:DIVision value.*

Sets or queries the vertical scale (the amount of time in each line) in the spectrogram. The vertical axis is composed of successive spectral displays. The new spectra can be added at a timed rate specified by this command. For example, if you set the scale to -5, one line is displayed every 5 spectra.

Conditions Measurement views: Spectrogram

Group Display commands

Syntax
```
DISPlay:SGRam:TIME:SCAle <value>
DISPlay:SGRam:TIME:SCAle?
```

Related Commands DISPlay:SGRam:TIME:OFFSet

Arguments `<value> ::  <NR1>` specifies the vertical scale.
Range: -1023 to 0. Zero (0) displays every spectrum.

Examples `DISPLAY:SGRAM:TIME:SCALE -5` displays one line every 5 spectra in the spectrogram.

DISPlay:SGRam:TIME:SCALe:PER:DIVision

Sets or queries the vertical Time/division value. Note that changing this value may change the DISPlay:SGRam:TIME:OVERlap:PERCent value.

Conditions Measurement views: Spectrogram

Group Display commands

Syntax
```
DISPlay:SGRam:TIME:SCALe:PER:DIVision <value>
DISPlay:SGRam:TIME:SCALe:PER:DIVision?
```

Related Commands DISPlay:SGRam:TIME:OVERlap:PERCent

Arguments <value> :: <NRf> specifies the time per division in seconds.

Examples DISPLAY:SGRAM:TIME:SCALE:PER:DIVISION? might return 1.1160000000E-3.

DISPlay:SGRam:TIME:SPECtrums:PERLine? (Query Only)

Queries the Spectrums/Line value.

Conditions Measurement views: Spectrogram

Group Display commands

Syntax DISPlay:SGRam:TIME:SPECtrums:PERLine?

Examples DISPLAY:SGRAM:TIME:SPECTRUMS:PERLINE? might return a 1.0000000000 as a response.

DISPlay:SPECtrum:FREQuency:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the spectrum view.

Conditions Measurement views: Spectrum

Group Display commands

Syntax DISPlay:SPECtrum:FREQuency:AUTO

Arguments None

Examples DISPLAY:SPECTRUM:FREQUENCY:AUTO rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:SPECtrum:FREQuency:OFFSet

Sets or queries the frequency offset (the value at the center of the horizontal axis) in the spectrum graph.

Conditions Measurement views: Spectrum

Group Display commands

Syntax

```
DISPlay:SPECtrum:FREQuency:OFFSet <value>
DISPlay:SPECtrum:FREQuency:OFFSet?
```

Related Commands DISPlay:SPECtrum:FREQuency[:SCAle]

Arguments <value> :: <NRf> specifies the frequency offset.
Range: [(center frequency) - (X scale) × 0.9] to [(center frequency) + (X scale) × 0.9]

Examples DISPLAY:SPECTRUM:FREQUENCY:OFFSET 1.45GHz sets the frequency offset to 1.45 GHz in the spectrum.

DISPlay:SPECtrum:FREQuency[:SCAle]

Sets or queries the horizontal range of the spectrum graph.

Conditions Measurement views: Spectrum

Group Display commands

Syntax

```
DISPlay:SPECtrum:FREQuency[:SCAle] <value>
DISPlay:SPECtrum:FREQuency[:SCAle]?
```

Related Commands DISPlay:SPECtrum:FREQuency:OFFSet

Arguments <value> :: <NRf> specifies the horizontal range.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `DISPLAY:SPECTRUM:FREQUENCY:SCALE 10MHz` sets the horizontal range to 10 MHz.

DISPlay:SPECtrum:MARKer:NOISe:MODE

Determines whether to enable or disable the marker noise mode in the spectrum view. In this mode, the marker readout indicates amplitude in dBm/Hz. It is valid for all markers except for the reference marker.

NOTE. *To use the marker noise mode, select dBm as the power unit by the [SENSe]:POWer:UNITs command.*

Conditions Measurement views: Spectrum

Group Display commands

Syntax
```
DISPlay:SPECtrum:MARKer:NOISe:MODE { OFF | ON | 0 | 1 }
DISPlay:SPECtrum:MARKer:NOISe:MODE?
```

Related Commands [SENSe]:POWer:UNITs

Arguments `OFF` or `0` disables the marker noise mode.

`ON` or `1` enables the marker noise mode.

Examples `DISPLAY:SPECTRUM:MARKER:NOISE:MODE ON` enables the marker noise mode.

DISPlay:SPECtrum:SCALe:LOG:STATe

Determines whether or not to set the horizontal axis logarithmic in the Spectrum view.

Conditions Measurement views: Spectrum

Group Display commands

Syntax
```
DISPlay:SPECtrum:SCALe:LOG:STATe { OFF | ON | 0 | 1 }
DISPlay:SPECtrum:SCALe:LOG:STATe?
```

Related Commands DISPlay:SPECtrum:X:LABel

Arguments `OFF` or `0` sets the horizontal axis linear (default).

`ON` or `1` sets the horizontal axis logarithmic.

Executing DISPlay:SPECtrum:SCALe:LOG:STATe ON sets DISPlay:SPECtrum:X:LABel SSFReq.

Examples `DISPLAY:SPECTRUM:SCALE:LOG:STATE ON` sets the horizontal axis logarithmic in the Spectrum view.

DISPlay:SPECtrum:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions Measurement views: Spectrum

Group Display commands

Syntax
```
DISPlay:SPECtrum:WINDow:TRACe:GRATicule:GRID:STATe { OFF |
ON | 0 | 1 }
DISPlay:SPECtrum:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples `DISPLAY:SPECTRUM:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the screen in the spectrum measurement.

DISPlay:SPECtrum:WINDow:TRACe:LEGend:STATe

Determines whether to show or hide the trace legend in the Spectrum view. The legend indicates the trace detection and function on the screen for each displayed spectrum trace.

Conditions Measurement views: Spectrum

Group Display commands

Syntax

```
DISPlay:SPECtrum:WINDow:TRACe:LEGend:STATe { OFF | ON | 0 | 1 }
DISPlay:SPECtrum:WINDow:TRACe:LEGend:STATe?
```

Arguments `OFF` or `0` hides the trace legend.

`ON` or `1` shows the trace legend.

Examples `DISPLAY:SPECTRUM:WINDOW:TRACE:LEGEND:STATE ON` shows the trace legend on the screen in the spectrum measurement.

DISPlay:SPECtrum:X:LABel

Sets or queries the labels for the horizontal (X) axis in the Spectrum view. The labels are indicated beneath the spectrum graph on the screen.

Conditions Measurement views: Spectrum

Group Display commands

Syntax

```
DISPlay:SPECtrum:X:LABel { SSFReq | CFSPan }
DISPlay:SPECtrum:X:LABel?
```

Arguments `SSFReq` sets the labels to the start and stop frequencies.

`CFSPan` sets the labels to the center frequency and span.

Examples `DISPLAY:SPECTRUM:X:LABEL SSFReq` sets the labels to the start and stop frequencies for the horizontal azis in the Spectrum view.

DISPlay:SPECtrum:Y[:SCALe]

Sets or queries the vertical range of the spectrum graph.

Conditions Measurement views: Spectrum

Group Display commands

Syntax

```
DISPlay:SPECtrum:Y[:SCALe] <value>
DISPlay:SPECtrum:Y[:SCALe]?
```

Arguments `<value> :: <NRf>` specifies the vertical range. Range: 0.1 to 200 dB. The amplitude unit is set by the [SENSe]:POWer:UNITs command.

You can omit the unit in the argument. When you include the unit in the argument, only dB is available. For the amplitude units Watts, Volts, and Amps, omit the unit in the argument.

Examples `DISPLAY:SPECTRUM:Y:SCALE 50dB` sets the vertical range to 50 dB in the Spectrum view.

DISPlay:SPECtrum:Y[:SCALe]:AUTO (No Query Form)

Rescales the vertical axis automatically to fit the waveform to the screen in the Spectrum view.

Conditions Measurement views: Spectrum

Group Display commands

Syntax

```
DISPlay:SPECtrum:Y[:SCALe]:AUTO
```

Arguments None

Examples `DISPLAY:SPECTRUM:Y:SCALE:AUTO` rescales the vertical scale automatically to fit the waveform to the screen.

DISPlay:SPECtrum:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the top edge of the vertical axis) of the spectrum graph.

Conditions Measurement views: Spectrum

Group Display commands

Syntax

```
DISPlay:SPECtrum:Y[:SCALe]:OFFSet <value>
DISPlay:SPECtrum:Y[:SCALe]:OFFSet?
```

Arguments `<value> ::` `<NRf>` specifies the vertical offset. Range: -270 to +150 dBm. The amplitude unit is set by the [SENSe]:POWer:UNITs command (default: dBm).

You can omit the unit in the argument. When you include the unit in the argument, only dBm is allowed. For the other amplitude units, omit the unit in the argument.

Examples `DISPLAY:SPECTRUM:Y:SCALE:OFFSET -12.5dBm` sets the vertical offset to -12.5 dBm.

DISPlay:SPECtrum:Y[:SCALe]:PDIVision

Sets or queries the vertical scale (per division) of the spectrum graph.

Conditions Measurement views: Spectrum

Group Display commands

Syntax

```
DISPlay:SPECtrum:Y[:SCALe]:PDIVision <value>
DISPlay:SPECtrum:Y[:SCALe]:PDIVision?
```

Related Commands [SENSe]:POWer:UNITs

Arguments `<value> ::` `<NRf>` specifies the vertical scale (per division). Range: 0.01 to 20 dB/div.

Examples `SENSE:SPECTRUM:Y:SCALE:PDIVISION 0.5` sets the vertical scale to 0.5 dB/div.

DISPlay:SPECtrum:Y[:SCALe]:RESet (No Query Form)

Resets the vertical scale of the spectrum graph to the default values: Vertical offset = Reference level and Vertical scale = 100 dB

Conditions Measurement views: Spectrum

Group Display commands

Syntax `DISPlay:SPECtrum:Y[:SCALe]:RESet`

Arguments None

Examples `DISPLAY:SPECTRUM:Y:SCALE:RESET` resets the vertical scale to the default values in the Spectrum view.

DISPlay:SPURious:MARKer:SHOW:STATe

Determines whether to show or hide the readout for the selected marker in the Spurious view.

Conditions Measurement views: Spurious

Group Display commands

Syntax
```
DISPlay:SPURious:MARKer:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:SPURious:MARKer:SHOW:STATe?
```

Arguments `OFF` or `0` hides the readout for the selected marker in the graph.

`ON` or `1` shows the readout for the selected marker in the graph.

Examples `DISPLAY:SPURIOUS:MARKER:SHOW:STATE ON` shows the readout for the selected marker in the graph.

DISPlay:SPURious:RESet:SCALe (No Query Form)

Resets the horizontal and vertical scale to the default values described below in the Spurious view.

Vertical offset = 0 dBm,
Vertical scale = 100 dB,
Horizontal offset = Center frequency, and
Horizontal scale = Default span

Conditions Measurement views: Spurious

Group Display commands

Syntax

```
DISPlay:SPURious:RESet:SCALe
```

Arguments None

Examples `DISPLAY:SPURIOUS:RESET:SCALE` resets the horizontal and vertical scale to the default values.

DISPlay:SPURious:SCALe:LOG:STATe

Determines whether or not to set the horizontal axis logarithmic in the Spurious view.

Conditions Measurement views: Spurious

Group Display commands

Syntax

```
DISPlay:SPURious:SCALe:LOG:STATe { OFF | ON | 0 | 1 }
DISPlay:SPURious:SCALe:LOG:STATe?
```

Arguments `OFF` or `0` sets the horizontal axis linear (default).

`ON` or `1` sets the horizontal axis logarithmic.

Examples `DISPLAY:SPURIOUS:SCALE:LOG:STATE ON` sets the horizontal axis logarithmic in the Spurious view.

DISPlay:SPURious:SELect:NUMBer

Sets or queries the spurious number in the Spurious view.

Conditions Measurement views: Spurious

Group Display commands

Syntax

```
DISPlay:SPURious:SELect:NUMBer <number>
DISPlay:SPURious:SELect:NUMBer?
```

Arguments `<number> ::  <NR1>` specifies the spurious number.
Range: 1 to the number of spurious signals.

Use the FETCh:SPURious:COUNt? or READ:SPURious:COUNt? query to get the number of spurious signals.

Examples `DISPLAY:SPURIOUS:SELECT:NUMBER 7` selects the spurious #7.

DISPlay:SPURious:SHOW:LIMit

Sets or queries how to display the limits.

Conditions Measurement views: Spurious

Group Display commands

Syntax

```
DISPlay:SPURious:SHOW:LIMit { SHADed | LINE | OFF }
DISPlay:SPURious:SHOW:LIMit?
```

Arguments `SHADed` displays the limits with shade.

`LINE` displays the limits with line only.

`OFF` hides the limits.

Examples `DISPLAY:SPURIOUS:SHOW:LIMIT LINE` displays the limits with line only.

DISPlay:SPURious:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions Measurement views: Spurious

Group Display commands

Syntax

```
DISPlay:SPURious:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:SPURious:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples `DISPLAY:SPURIOUS:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the screen.

DISPlay:SPURious:X[:SCALe]:AUTO (No Query Form)

Rescales the horizontal axis automatically to fit the waveform to the screen in the Spurious view.

Conditions Measurement views: Spurious

Group Display commands

Syntax

```
DISPlay:SPURious:X[:SCALe]:AUTO
```

Arguments None

Examples `DISPLAY:SPURIOUS:X:SCALE:AUTO` rescales the horizontal scale automatically to fit the waveform to the screen.

DISPlay:SPURious:X[:SCALe]:STARt

Sets or queries the minimum horizontal value (left edge) of the spectrum graph in the Spurious view.

Conditions Measurement views: Spurious

Group Display commands

Syntax
```
DISPlay:SPURious:X[:SCALe]:STARt <value>
DISPlay:SPURious:X[:SCALe]:STARt?
```

Arguments `<value> ::` `<NRf>` specifies the minimum horizontal value.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `DISPLAY:SPURIOUS:X:SCALE:START 1.61GHz` sets the minimum horizontal value to 1.61 GHz in the spectrum graph.

DISPlay:SPURious:X[:SCALe]:STOP

Sets or queries the maximum horizontal value (right edge) of the spectrum graph in the Spurious view.

Conditions Measurement views: Spurious

Group Display commands

Syntax
```
DISPlay:SPURious:X[:SCALe]:STOP <value>
DISPlay:SPURious:X[:SCALe]:STOP?
```

Arguments `<value> ::` `<NRf>` specifies the minimum horizontal value.
Range:

- RSA5103AB – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `DISPLAY:SPURIOUS:X:SCALE:STOP 2.16GHz` sets the maximum horizontal value to 2.16 GHz in the spectrum graph.

DISPlay:SPURious:Y[:SCALe]

Sets or queries the vertical range of the spectrum graph in the Spurious view.

Conditions Measurement views: Spurious

Group Display commands

Syntax
```
DISPlay:SPURious:Y[:SCALe] <value>
DISPlay:SPURious:Y[:SCALe]?
```

Arguments `<value> ::= <NRf>` specifies the vertical range. Range: 0.1 to 200 dB. The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `DISPLAY:SPURIOUS:Y:SCALE 100` sets the vertical range to 100 dB in the Spurious view.

DISPlay:SPURious:Y[:SCALe]:AUTO (No Query Form)

Rescales the vertical axis automatically to fit the waveform to the screen in the Spurious view.

Conditions Measurement views: Spurious

Group Display commands

Syntax `DISPlay:SPURious:Y[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:SPURIOUS:Y:SCALE:AUTO` rescales the vertical scale automatically to fit the waveform to the screen.

DISPlay:SPURious:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the top edge of the vertical axis) of the spectrum graph in the Spurious view.

Conditions Measurement views: Spurious

Group Display commands

Syntax

```
DISPlay:SPURious:Y[:SCALe]:OFFSet <value>
DISPlay:SPURious:Y[:SCALe]:OFFSet?
```

Arguments `<value> ::= <NRf>` specifies the vertical offset. Range: -270 to +150 dBm. The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `DISPLAY:SPURIOUS:Y:SCALE:OFFSET -12.5` sets the vertical offset to -12.5 dBm in the spectrum graph.

DISPlay:TDIagram:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions Measurement views: Trellis diagram

Group Display commands

Syntax

```
DISPlay:TDIagram:WINDow:TRACe:GRATicule:GRID:STATe { OFF |
ON | 0 | 1 }
DISPlay:TDIagram:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples `DISPLAY:TDIAGRAM:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the screen in the trellis diagram.

DISPlay:TDIagram:Y[:SCALe]

Sets or queries the vertical range of the trellis diagram.

Conditions Measurement views: Trellis diagram

Group Display commands

Syntax
```
DISPlay:TDIagram:Y[:SCALe] <value>
DISPlay:TDIagram:Y[:SCALe]?
```

Arguments <value> :: <NRf> specifies the vertical range. Range: 1 ° to 1 T°.

Examples DISPLAY:TDIAGRAM:Y:SCALE 2.5 sets the vertical range to 2.5 ° in the trellis diagram.

DISPlay:TDIagram:Y[:SCALe]:AUTO (No Query Form)

Sets the vertical scale automatically to fit the waveform to the screen in the trellis diagram.

Conditions Measurement views: Trellis diagram

Group Display commands

Syntax DISPlay:TDIagram:Y[:SCALe]:AUTO

Arguments None

Examples DISPLAY:TDIAGRAM:Y:SCALE:AUTO sets the vertical scale automatically to fit the waveform to the screen.

DISPlay:TDIagram:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (center point of the vertical axis) of the Trellis diagram.

Conditions Measurement views: Trellis diagram

Group Display commands

Syntax
```
DISPlay:TDIagram:Y[:SCALe]:OFFSet <value>
DISPlay:TDIagram:Y[:SCALe]:OFFSet?
```

Arguments `<value> ::` `<NRf>` specifies the vertical offset. Range: -1 T° to +1 T°.

Examples `DISPLAY:TDIAGRAM:Y:SCALE:OFFSET -28.5` sets the vertical offset to -28.5 ° in the Trellis diagram.

DISPlay:TG:MEASview:DELete TXGain (No Query Form)

This command removes the Transmission Gain display icon from the Selected displays filed of the Selected Displays window.

Conditions Measurement view: Select Displays window

Group Display group

Syntax `DISPlay:TG:MEASview:DELete TXGain`

Arguments None

Examples `DISPLAY:TG:MEASVIEW:DELETE TXGAIN`

DISPlay:TG:MEASview:NEW TXGain (No Query Form)

This command opens a new Transmission Gain measurement display window.

Conditions Measurement view: Select Displays window

Group Display group

Syntax `DISPlay:TG:MEASview:NEW TXGain`

Related Commands

Arguments

Examples

```
DISPLAY:TG:MEASVIEW:NEW TXGAIN
```

DISPlay:TG:MEASview:SELect TXGain

This command selects the Transmission Gain display icon in the Select Display window and queries if the display is selected or not.

Conditions Measurement view: Select Displays window

Group Display group

Syntax

```
DISPlay:TG:MEASview:SELect TXGain { OFF | ON | 0 | 1 }
DISPlay:TG:MEASview:SELect TXGain?
```

Arguments `OFF` or `0` deselects the Transmission Gain display icon.

`ON` or `1` selects the Transmission Gain display icon.

Returns 0 means the icon is not selected.

1 means the icon is selected.

Examples

```
DISPLAY:TG:MEASVIEW:SELECT TXGAIN
```

DISPlay:TOVerview:WINDow:NAVigator:STATe

Sets or queries whether the navigator view of the Time Overview display is on or off.

Conditions Measurement view: Time Overview

Group Display commands

Syntax

```
DISPlay:TOVerview:WINDow:NAVigator:STATe { OFF | ON | 0 |
1 }
DISPlay:TOVerview:WINDow:NAVigator:STATe?
```

Arguments

OFF or 0 sets the Time Overview display to not show in Navigator View mode.

ON or 1 sets the Time Overview display to show in Navigator View mode.

Returns

OFF or 0 means that the Time Overview display is not in Navigator View mode.

ON or 1 means that the Time Overview display is in Navigator View mode.

Examples

DISPLAY:TOVERVIEW:WINDOW:NAVIGATOR:STATE ON will set the Time Overview display to show in Navigator View mode.

DISPlay:TOVerview:WINDow:TIME:MODE

Sets or queries the type of time analysis to be performed for the Time Overview display.

Conditions

Measurement view: Time Overview

Group

Display commands

Syntax

```
DISPlay:TOVerview:WINDow:TIME:MODE { ANALysis | SPECtrum |
LINKed }
DISPlay:TOVerview:WINDow:TIME:MODE?
```

Related Commands

Arguments

ANALysis sets the type of analysis time to Analysis length, which is the period of time within the acquisition record over which all other measurements (such as Amplitude vs. Time) are made.

SPECtrum sets the type of analysis time to Spectrum length, which is the period of time within the acquisition record over which the spectrum is calculated.

LINKed sets the type of analysis time to lock the Spectrum length and Analysis length together so that the data used to produce the Spectrum display is also used for measurement displays.

Returns

ANAL means that the type of time analysis is set to Analysis.

SPEC means that the type of time analysis is set to Spectrum.

LINK means that the type of time analysis is set to Linked.

Examples `DISPLAY:TOVERVIEW:WINDOW:TIME:MODE?` might return SPEC, which indicates that the type of analysis time is set to Spectrum, which is the period of time within the acquisition record over which the spectrum is calculated.

DISPlay:TOVerview:WINDow:TRACe:GRATicule:GRID:STATe

Determines whether to show or hide the graticule grid on the screen.

Conditions Measurement views: Time overview

Group Display commands

Syntax
```
DISPlay:TOVerview:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:TOVerview:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `OFF` or `0` hides the graticule grid.

`ON` or `1` shows the graticule grid.

Examples `DISPLAY:TOVERVIEW:WINDOW:TRACE:GRATICULE:GRID:STATE ON` shows the graticule grid on the screen in the time overview.

DISPlay:TOVerview:WINDow:TRACe:LEGend:STATe

Sets or queries the trace legend state in the Time Overview display to on (showing) or off (hidden).

Conditions Measurement view: Time Overview

Group Display commands

Syntax
```
DISPlay:TOVerview:WINDow:TRACe:LEGend:STATe { OFF | ON | 0 | 1 }
DISPlay:TOVerview:WINDow:TRACe:LEGend:STATe?
```

Related Commands

Arguments `OFF` or `0` hides the trace legend in the Time Overview display.

`ON` or `1` shows the trace legend in the Time Overview display.

Returns `0` means the trace legend in the Time Overview display is hidden (off).

`1` means the trace legend in the Time Overview display is showing (on).

Examples `DISPLAY:TOVERVIEW:WINDOW:TRACE:LEGEND:STATE ON` will turn on the trace legend in the Time Overview display.

`DISPLAY:TOVERVIEW:WINDOW:TRACE:LEGEND:STATE? 1` indicates that the trace legend in the Time Overview display is showing (on).

DISPlay:TOVerview:X[:SCALe]

Sets or queries the horizontal scale (full-scale time) of the time overview.

Conditions Measurement views: Time overview

Group Display commands

Syntax
```
DISPlay:TOVerview:X[:SCALe] <value>
DISPlay:TOVerview:X[:SCALe]?
```

Related Commands DISPlay:TOVerview:X[:SCALe]:OFFSet

Arguments `<value> :: <NRf>` specifies the horizontal scale in full-scale time. Range: 10 ns to the acquisition memory capacity.

You can see the acquisition memory capacity using the [SENSe]:ACQuisition:MEMory:CAPacity[:TIME]? query.

Examples `DISPLAY:TOVERVIEW:X:SCALE 12.5us` sets the horizontal scale to 12.5 µs.

DISPlay:TOVerview:X[:SCALe]:AUTO (No Query Form)

Sets the horizontal scale and offset automatically to fit the waveform to the screen in the time overview.

Conditions Measurement views: Time overview

Group Display commands

Syntax `DISPlay:TOVerview:X[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:TOVERVIEW:X:SCALE:AUTO` sets the horizontal scale and offset automatically to fit the waveform to the screen.

DISPlay:TOVerview:X[:SCALe]:OFFSet

Sets or queries the minimum horizontal value (left edge) of the time overview.

Conditions Measurement views: Time overview

Group Display commands

Syntax
```
DISPlay:TOVerview:X[:SCALe]:OFFSet <value>
DISPlay:TOVerview:X[:SCALe]:OFFSet?
```

Related Commands DISPlay:TOVerview:X[:SCALe]

Arguments `<value> :: <NRf>` specifies the minimum horizontal value.
Range: [(analysis offset) - (X scale) × 0.9] to [(analysis offset) + (analysis length) - (X scale) × 0.1]

Examples `DISPLAY:TOVERVIEW:X:SCALE:OFFSET 800ns` sets the minimum horizontal value to 800 ns in the time overview.

DISPlay:TOVerview:Y[:SCALe]

Sets or queries the vertical range of the time overview.

Conditions Measurement views: Time overview

Group Display commands

Syntax
```
DISPlay:TOVerview:Y[:SCALe] <value>
DISPlay:TOVerview:Y[:SCALe]?
```

Related Commands DISPlay:TOVerview:Y[:SCALe]:OFFSet

Arguments `<value> :: <NRf>` specifies the vertical range. Range: 0.1 to 200 dB.

Examples `DISPLAY:TOVERVIEW:Y:SCALE 50` sets the vertical range to 50 dBm in the time overview.

DISPlay:TOVerview:Y[:SCALe]:AUTO (No Query Form)

Sets the vertical scale and offset automatically to fit the waveform to the screen in the time overview.

Conditions Measurement views: Time overview

Group Display commands

Syntax `DISPlay:TOVerview:Y[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:TOVERVIEW:Y:SCALE:AUTO` sets the vertical scale and offset automatically to fit the waveform to the screen.

DISPlay:TOVerview:Y[:SCALe]:OFFSet

Sets or queries the vertical offset (the value at the top edge of the vertical axis) in the time overview.

Conditions Measurement views: Time overview

Group Display commands

Syntax
```
DISPlay:TOVerview:Y[:SCALe]:OFFSet <value>
DISPlay:TOVerview:Y[:SCALe]:OFFSet?
```

Related Commands DISPlay:TOVerview:Y[:SCALe]

Arguments `<value> ::= <NRf>` specifies the vertical offset. Range: -170 to +50 dBm.

Examples `DISPLAY:TOVERVIEW:Y:SCALE:OFFSET -80` sets the vertical offset to -80 dBm in the time overview.

DISPlay:TOVerview:Y[:SCALe]:RESCale (No Query Form)

Sets the vertical scale automatically to fit the waveform to the screen in the time overview.

Conditions Measurement views: Time overview

Group Display commands

Syntax `DISPlay:TOVerview:Y[:SCALe]:RESCale`

Arguments None

Examples `DISPLAY:TOVERVIEW:Y:SCALE:RESCALE` sets the vertical scale automatically to fit the waveform to the screen.

DISPlay:TXGain:MARKer:SHOW:STATe

Displays or hides the marker readout, but not the marker itself, on the graph area.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax
```
DISPlay:TXGain:MARKer:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:TXGain:MARKer:SHOW:STATe?
```

Arguments OFF or 0 hides the readout for the selected marker.

ON or 1 shows the readout for the selected marker.

Returns OFF or 0 means that the readout for the selected marker is hidden.

ON or 1 means that the readout for the selected marker is showing.

Examples `DISPLAY:TXGAIN:MARKER:SHOW:STATE ON` shows the readout for the marker on the display.

DISPlay:TXGain[:SCALe]:AUTO (No Query Form)

Automatically rescales the horizontal and vertical scale.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax `DISPlay:TXGain[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:TXGAIN[:SCALE]:AUTO` rescales the horizontal and vertical scale automatically.

DISPlay:TXGain:WINDow:TRACe:GRATicule:GRID:STATe

Displays or hides the graticule grid in the display. Queries whether the graticule grid is hidden or showing.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax
```
DISPlay:TXGain:WINDow:TRACe:GRATicule:GRID:STATe { OFF | ON | 0 | 1 }
DISPlay:TXGain:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments OFF or 0 hides the graticule.

ON or 1 shows the graticule.

Returns OFF or 0 means that the graticule is hidden.

ON or 1 means that the graticule is showing.

Examples `DISPLAY:TXGAIN:WINDOW:TRACE:GRATICULE:GRID:STATE OFF` will hide the graticule.

DISPlay:TXGain:WINDow:TRACe:POINts:SHOW:STATe

Sets to mark each measurement point on the trace.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax
```
DISPlay:TXGain:WINDow:TRACe:POINts:SHOW:STATe { OFF | ON | 0 | 1 }
DISPlay:TXGain:WINDow:TRACe:POINts:SHOW:STATe?
```

Arguments OFF or 0 hides each measurement point on the trace.

ON or 1 shows each measurement point on the trace.

Returns OFF or 0 means that each measurement point on the trace is hidden.

ON or 1 means that each measurement point on the trace is showing.

Examples `DISPLAY:TXGAIN:WINDOW:TRACE:POINTS:SHOW:STATE 1` shows each measurement point on the trace.

DISPlay:TXGain:X[:SCALe]:LOG:STATe

Sets or queries the display to show the frequency axis in a logarithmic scale.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax
```
DISPlay:TXGain:X[:SCALe]:LOG:STATe { OFF | ON | 0 | 1 }
DISPlay:TXGain:X[:SCALe]:LOG:STATe?
```

Arguments OFF or 0 sets the frequency axis in linear scale.

ON or 1 sets the frequency axis in logarithmic scale.

Returns OFF or 0 means that the frequency axis is set to linear scale.

ON or 1 means that the frequency axis is set to logarithmic scale.

Examples `DISPLAY:TXGAIN:X:SCALE:LOG:STATE 1` sets the frequency axis to logarithmic scale.

DISPlay:TXGain:X[:SCALe]:STARt

Set the Zoom Start frequency. This only affects the start frequency shown on the graph.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax
```
DISPlay:TXGain:X[:SCALe]:STARt <value>
DISPlay:TXGain:X[:SCALe]:STARt?
```

Arguments <value>::=<NRf> is a floating point number that is the zoom start frequency.

Returns See Arguments.

Examples `DISPLAY:TXGAIN:X:SCALE:START 1E9` sets the zoom start frequency to 1 GHz.

DISPlay:TXGain:X[:SCALe]:AUTO (No Query Form)

Resets the scale of the horizontal axis to contain the complete trace.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax `DISPlay:TXGain:X[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:TXGAIN:X:SCALE:AUTO` resets the scale of the horizontal axis to contain the complete trace.

DISPlay:TXGain:X[:SCALe]:STOP

Set the Zoom Stop frequency. This only affects the stop frequency shown on the graph.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax
```
DISPlay:TXGain:X[:SCALe]:STOP <value>
DISPlay:TXGain:X[:SCALe]:STOP?
```

Arguments <value>::=<NRf> specifies the zoom stop frequency.

Returns A floating point value that is the zoom stop frequency.

Examples `DISPLAY:TXGAIN:X:SCALE:STOP 1E9` sets the zoom stop frequency to 1 GHz.

DISPlay:TXGain:Y[:SCALe]:BOTTom

Sets or queries the bottom of the vertical position of the display.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax
```
DISPlay:TXGain:Y[:SCALe]:BOTTom <value>
DISPlay:TXGain:Y[:SCALe]:BOTTom?
```

Arguments <value>::=<NRf> specifies the bottom of the vertical position of the display.

Returns A floating point value that is the bottom of the vertical position of the display.

Examples `DISPLAY:TXGAIN:Y:SCALE:BOTTOM –100` sets the bottom of Y axis to -100 dB.

DISPlay:TXGain:Y[:SCALe]:PDIVision

Sets or queries the vertical scale (per division) of the graph.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax
```
DISPlay:TXGain:Y[:SCALe]:PDIVision <value>
DISPlay:TXGain:Y[:SCALe]:PDIVision?
```

Arguments <value>::=<NRf> specifies the vertical scale (per division).

Returns A floating point value that is the vertical scale (per division) of the graph.

Examples `DISPLAY:TXGAIN:Y:SCALE:PDIVision 0.5` sets the vertical scale to 0.5 dB/div.

DISPlay:TXGain:Y[:SCALe]:POSition

Sets or queries the vertical position of the trace.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax
```
DISPlay:TXGain:Y[:SCALe]:POSition <value>
DISPlay:TXGain:Y[:SCALe]:POSition?
```

Arguments <value>::=<NRf> specifies the vertical position.

Returns A floating point value that is the vertical position.

Examples `DISPLAY:TXGAIN:Y:SCALE:PDIVision 0` sets the vertical position to 0 dB.

DISPlay:TXGain:Y[:SCALe]:TOP

Sets or queries the top of the vertical scale.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax
```
DISPlay:TXGain:Y[:SCALe]:TOP <value>
DISPlay:TXGain:Y[:SCALe]:TOP?
```

Arguments <value>::=<NRf> specifies the top of the vertical scale.

Returns A floating point value that is the top of the vertical scale.

Examples `DISPLAY:TXGAIN:Y:SCALE:TOP 10` sets the vertical top to 10 dB.

DISPlay:TXGain:Y[:SCALe]:AUTO (No Query Form)

Automatically scales the plot vertically.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax `DISPlay:TXGain:Y[:SCALe]:AUTO`

Arguments None

Examples `DISPLAY:TXGAIN:Y:SCALE:AUTO` automatically scales the plot vertically.

DISPlay:TXGain:Y[:SCALe]

Sets or queries the vertical scale value in dB.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax
```
DISPlay:TXGain:Y[:SCALe] <value>
DISPlay:TXGain:Y[:SCALe]?
```

Arguments <value>::=<NRf> specifies the vertical scale in dB.

Returns A floating point value that is the vertical scale in dB.

Examples `DISPLAY:TXGAIN:Y:SCALE 100` sets the vertical scale to 100 dB.

DISPlay:WINDow:ACTive:MEASurement? (Query Only)

Queries the active measurement views.

Conditions Measurement views: All

Group Display commands

Syntax
```
DISPlay:WINDow:ACTive:MEASurement?
```

Arguments None

Returns
```
<view1>,<view2>,...,<view(n)>
```

Where
`<view(n)> ::  <string>` is the view name as shown in the following table.

Table 2-25: Measurement view mnemonic

Return value	Measurement view	Display group
“SPEC	Spectrum	General signal viewing
“DPX”	DPX (Digital Phosphor) spectrum	
“MAGVT”	Amplitude versus Time	
“ACP”	Channel power and ACPR	
“FVT”	Frequency versus Time	
“PHVT”	Phase versus Time	
“IQVT”	RF I&Q versus Time	
“SGRam”	Spectrogram	
“TOV”	Time overview	
“AM”	Amplitude modulation	Analog Modulation
“FM”	Frequency modulation	
“PM”	Phase modulation	
“CONS”	Constellation	General purpose digital modulation
“DIQV”	Demodulated I&Q versus Time	
“EDI”	Eye Diagram	
“EVM”	EVM versus Time	
“FVDT”	Frequency deviation versus Time	
“MERR”	Magnitude error versus Time	
“PERR”	Phase error versus Time	
“SIGN”	Signal quality	
“STAB”	Symbol table	
“TDI”	Trellis Diagram	
“CCDF”	CCDF	RF measurements
“MCP”	MCPR (Multiple Carrier Power Ratio)	
“OBW”	Occupied bandwidth	
“PNO”	Phase noise	
“SPUR”	Spurious	
“STAT”	Pulse statistics	Pulsed RF
“RES”	Pulse table (results table)	
“TRAC”	Pulse trace	

Examples `DISPLAY:WINDOW:ACTIVE:MEASUREMENT?` might return `"SPEC","TRAC"`, indicating that the views of spectrum and pulse trace are displayed on the screen.

DISPlay:WINDow:COLor:SCHeme

Sets or queries the color scheme for displaying traces and background on the screen.

Conditions Measurement views: All

Group Display commands

Syntax

```
DISPlay:WINDow:COLor:SCHeme { THUNderstorm | BLIZzard |
CLASsic }
DISPlay:WINDow:COLor:SCHeme?
```

Arguments

`THUNderstorm` displays the backgound in dark blue.

`BLIZzard` displays the backgound in white. It saves ink when printing the screen image.

`CLASsic` displays the backgound in black (default).

Examples `DISPLAY:WINDOW:COLOR:SCHEME BLIZzard` displays the backgound in white.

DISPlay:WINDow:OPTimized:MEASurement? (Query Only)

Queries the measurement views that are optimized. "Optimized" means that there is a perfect match between the view's settings and the actual acquisition parameters to meet the specifications. When multiple measurements are running at one time, the measurements can have different requirements for setting the acquisition hardware. You can make a measurement optimized by selecting it using the following commands:

- DISPlay:GENeral:MEASview:SELect for the general signal viewing
- DISPlay:DDEMod:MEASview:SELect for the digital modulation views
- DISPlay:GPRF:MEASview:SELect for the RF measurement views
- DISPlay:PULSe:MEASview:SELect for the pulsed RF measurement views

Conditions Measurement views: All

Group Display commands

Syntax `DISPlay:WINDow:OPTimized:MEASurement?`

Arguments None

Returns `<view1>,<view2>,...,<view(n)>`

Where
`<view(n)> ::  <string>` is the view name as shown in the table. (See Table 2-25 on page 2-690.)

Examples `DISPLAY:WINDOW:OPTIMIZED:MEASUREMENT?` might return `"SPEC","MCP"`, indicating that the views of spectrum and MCPR are optimized.

DISPlay:WLAN:CONSte[:SCALe]

Sets the magnification value (zoom factor) for the WLAN constellation plot.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:CONSte[:SCALe] <NRf>
DISPlay:WLAN:CONSte[:SCALe]?
```

Arguments Floating point number that represents the scale factor. Valid inputs are any floating point number between 0.1 and 10. A positive value zooms in and a negative value zooms out.

Returns A floating point number indicating the current magnification value.

Examples `DISP:WLAN:CONS:SCAL 5.0` zooms the display out by a factor of 5.

DISPlay:WLAN:CONSte:X:OFFSet

Sets or queries the value of the offset of the horizontal axis from the center of the WLAN Constellation display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:CONSte:X:OFFSet <NRf>
DISPlay:WLAN:CONSte:X:OFFSet?
```

Arguments Floating point number between limits which are set as a function of the current scale value guaranteed to keep the Constellation plot at least partially visible on the display. Positive inputs push the plot down (as viewed by the user) and negative inputs push the plot up. 0 recenters the plot.

Returns The value of the offset of the X axis from the center for the Constellation plot as a floating point number.

Examples `DISP:WLAN:CONST:X:OFFS 1` shifts the constellation display to the left by 1.

DISPlay:WLAN:CONSte:Y:OFFSet

Sets or queries the value of the offset of the vertical axis from the center for the WLAN Constellation display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:CONSte:Y:OFFSet <NRf>
DISPlay:WLAN:CONSte:Y:OFFSet?
```

Arguments Floating point number between limits which are set as a function of the current scale value guaranteed to keep the constellation plot at least partially visible on the display. Positive inputs push the plot down (as viewed by the user) and negative inputs push the plot up. 0 re-centers the plot.

Returns The value of the offset of the Y axis from the center for the Constellation display as a floating point number.

Examples `DISP:WLAN:CONST:Y:OFFS 1` shifts the constellation display up by 1.

DISPlay:WLAN:CRESponse:AUTO (No Query Form)

Automatically rescales the horizontal and vertical values for the best display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax `DISPlay:WLAN:CRESponse:AUTO`

Arguments None

Examples `DISP:WLAN:CRES:AUTO` automatically rescales the horizontal and vertical values.

DISPlay:WLAN:CRESponse:FREQuency:AUTO (No Query Form)

Automatically rescales the horizontal subcarrier (frequency) start and stop values to fit the waveform to the screen in the WLAN Channel Response display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax `DISPlay:WLAN:CRESponse:FREQuency:AUTO`

Arguments None

Examples `DISP:WLAN:CRES:FREQ:AUTO` automatically rescales the horizontal subcarrier (frequency) start and stop values to fit the waveform to the screen.

DISPlay:WLAN:CRESponse:FREQuency:STARt

Specifies or queries the horizontal start value (left edge) on the WLAN Channel Response display. The units are either Frequency (Hz) or Subcarriers, and are set using the command [SENSe]:WLAN:UNIT:FREQuency.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax

```
DISPlay:WLAN:CRESponse:FREQuency:STARt <NRf>
DISPlay:WLAN:CRESponse:FREQuency:STARt?
```

Arguments Floating point number that specifies the horizontal start value (left edge) on the graph.

Returns The returned value indicates the Subcarrier (for non-b standards only) or frequency shown at the left edge of the display.

Examples `DISP:WLAN:CRES:FREQ:START 2.41e9` sets the horizontal start value (left edge) to 2.41 GHz (assuming the units have been set to Frequency).

DISPlay:WLAN:CRESponse:FREQuency:STOP

Specifies or queries the horizontal stop value (right edge) on the WLAN Channel Response display. The units are either Frequency (Hz) or Subcarriers, and are set using the command [SENSe]:WLAN:UNIT:FREQuency.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax

```
DISPlay:WLAN:CRESponse:FREQuency:STOP <NRf>
DISPlay:WLAN:CRESponse:FREQuency:STOP?
```

Arguments Floating point number that specifies the horizontal stop value (right edge) on the display.

Examples `DISP:WLAN:CRES:FREQ:STOP 8.1250e9` sets the horizontal stop value to 8.1250 GHz (assuming the units have been set to Frequency).

DISPlay:WLAN:CRESponse:MAGNitude:AUTO (No Query Form)

Rescales the vertical magnitude offset (position) and scale values to automatically to fit the waveform to the screen in the WLAN Channel Response magnitude graph.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax `DISPlay:WLAN:CRESponse:MAGNitude:AUTO`

Arguments None

Examples `DISP:WLAN:CRES:MAGN:AUTO` rescales the vertical magnitude offset (position) and scale values automatically to fit the waveform to the screen.

DISPlay:WLAN:CRESponse:MAGNitude:OFFSet

Sets or queries the vertical offset (position in the center of the vertical axis) in the WLAN Channel Response magnitude graph.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:CRESponse:MAGNitude:OFFSet <NRf>
DISPlay:WLAN:CRESponse:MAGNitude:OFFSet?
```

Arguments Floating point number that specifies the vertical offset in dB.

Examples `DISP:WLAN:CRES:MAGN:OFFS 5` sets the vertical offset to 5 dB.

DISPlay:WLAN:CRESponse:MAGNitude[:SCALe]

Sets or queries the value of the vertical scale for the WLAN Channel Response magnitude display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:CRESponse:MAGNitude[:SCALe] <NRf>
DISPlay:WLAN:CRESponse:MAGNitude[:SCALe]?
```

Arguments Floating point number that represents the vertical scale value in dB.

Examples `DISP:WLAN:CRES:MAGN:SCAL 5.0` sets the vertical scale to 5 dB.

DISPlay:WLAN:CRESponse:MARKer:SHOW:STATe

Shows or hides the marker readout on the WLAN Channel Response display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:CRESponse:MARKer:SHOW:STATe {0|1|OFF|ON}
DISPlay:WLAN:CRESponse:MARKer:SHOW:STATe?
```

Arguments `ON or 1` specifies to show the marker readout.

`OFF or 0` specifies to turn the marker readout off.

Examples `DISP:WLAN:CRES:MARK:SHOW:STATE ON` specifies to turn the marker readout on.

DISPlay:WLAN:CRESponse:PHASe:AUTO (No Query Form)

Rescales the vertical phase value automatically to fit the waveform to the screen in the WLAN Channel Response display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax `DISPlay:WLAN:CRESponse:PHASe:AUTO`

Arguments None

Examples `DISP:WLAN:CRES:PHAS:AUTO` rescales the vertical phase value automatically to fit the waveform to the screen.

DISPlay:WLAN:CRESponse:PHASe:OFFSet

Sets or queries the vertical offset (position in the center of the vertical axis) in the WLAN Channel Response phase graph.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:CRESponse:PHASe:OFFSet <NRf>
DISPlay:WLAN:CRESponse:PHASe:OFFSet?
```

Arguments Floating point number that specifies the vertical offset (position) in degrees.

Examples `DISPLAY:WLAN:CRESPONSE:PHASE:OFFSET −14.5` sets the vertical offset value to -14.5º in the WLAN Channel Response phase graph.

DISPlay:WLAN:CRESponse:PHASe[:SCALe]

Sets or queries the phase value for the WLAN Channel Response display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:CRESponse:PHASe[:SCALe] <NRf>
DISPlay:WLAN:CRESponse:PHASe[:SCALe]?
```

Arguments Floating point number that represents the phase value in degrees.

Examples `DISP:WLAN:CRES:PHASE:SCAL 5` sets the vertical scale to 5 degrees for the display.

DISPlay:WLAN:CRESponse:WINDow:SELect:PLOT

Specifies or queries which graphs are displayed in the WLAN Channel Response display view. Select from Magnitude, Phase, or Both.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:CRESponse:WINDow:SELect:PLOT
{MAGNitude|PHASe|BOTH}
DISPlay:WLAN:CRESponse:WINDow:SELect:PLOT?
```

Arguments `MAGNitude` displays the Magnitude graph.

`PHASe` displays the Phase graph.

BOTH displays both graphs.

Examples DISP:WLAN:CRES:WIND:SEL:PLOT PHAS displays the Phase graph on the WLAN Channel Response analysis display.

DISPlay:WLAN:CRESponse:WINDow:TRACe:GRATicule:GRID:STATe

Sets or queries whether to show the graticule grid on the WLAN Channel Response analysis display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:CRESponse:WINDow:TRACe:GRATicule:GRID:STATe
{0|1|OFF|ON}
DISPlay:WLAN:CRESponse:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments ON or 1 specifies to show the graticule.

OFF or 0 specifies to turn it off.

Examples SENSE:WLAN:CRES:WIND:TRAC:GRAT:GRID:STATE ON specifies to turn the graticule on.

DISPlay:WLAN:EVM:FREQuency:AUTO (No Query Form)

Automatically rescales the horizontal subcarrier (frequency) start and stop values to fit the waveform to the screen in the WLAN EVM display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax DISPlay:WLAN:EVM:FREQuency:AUTO

Arguments None

Examples `DISP:WLAN:EVM:FREQ:AUTO` automatically rescales the horizontal Subcarrier (frequency) values to fit the waveform to the screen.

DISPlay:WLAN:EVM:FREQuency:RESet (No Query Form)

Automatically resets the horizontal subcarrier (frequency) start and stop values for the WLAN EVM display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax `DISPlay:WLAN:EVM:FREQuency:RESet`

Arguments None

Examples `DISP:WLAN:EVM:FREQ:RES` automatically resets the horizontal subcarrier start and stop values for the WLAN EVM display.

DISPlay:WLAN:EVM:FREQuency:STARt

Specifies or queries the horizontal start value (left edge) on the WLAN EVM display. The units are either Frequency (Hz) or Subcarriers, and are set using the command [SENSe]:WLAN:UNIT:FREQuency.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:EVM:FREQuency:STARt <NRf>
DISPlay:WLAN:EVM:FREQuency:STARt?
```

Arguments Floating point number that specifies the horizontal start value (left edge) on the graph.

Returns The returned value indicates the Subcarrier (for non-b standards only) or frequency shown at the left edge of the display.

Examples `DISP:WLAN:EVM:FREQ:START 2.41e9` sets the horizontal start value (left edge) to 2.41e9 MHz (assuming the units have been set to Frequency).

DISPlay:WLAN:EVM:FREQuency:STOP

Specifies or queries the horizontal stop value (right edge) on the WLAN EVM display. The units are either Frequency (Hz) or Subcarriers, and are set using the command [SENSe]:WLAN:UNIT:FREQuency.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:EVM:FREQuency:STOP <NRf>
DISPlay:WLAN:EVM:FREQuency:STOP?
```

Arguments Floating point number that specifies the horizontal stop value (right edge) on the display.

Examples `DISP:WLAN:EVM:FREQ:STOP 8.1250` sets the horizontal stop value to 8.1250 MHz (assuming the units have been set to Frequency).

DISPlay:WLAN:EVM:MARKer:SHOW:STATe

Shows or hides the Marker readout on the WLAN EVM display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax

```
DISPlay:WLAN:EVM:MARKer:SHOW:STATe {0|1|OFF|ON}
DISPlay:WLAN:EVM:MARKer:SHOW:STATe?
```

Arguments

`ON or 1` specifies to show the Marker readout.

`OFF or 0` specifies to turn the Marker readout off.

Examples

`DISP:WLAN:EVM:MARK:SHOW:STATE ON` specifies to turn the Marker readout on.

DISPlay:WLAN:EVM[:SCALe]:AUTO (No Query Form)

Automatically adjusts all vertical and horizontal values to provide the best display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:EVM[:SCALe]:AUTO
```

Arguments

None

Examples

`DISP:WLAN:EVM:SCALE:AUTO` automatically adjusts all vertical and horizontal values to provide the best display.

DISPlay:WLAN:EVM:TIME:AUTO (No Query Form)

Sets the horizontal start and stop values to fit the waveform to the WLAN EVM display, and also enables Auto horizontal scaling so the waveform will remain properly scaled as burst lengths change.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax `DISPlay:WLAN:EVM:TIME:AUTO`

Arguments None

Examples `DISP:WLAN:EVM:TIME:AUTO` sets the horizontal start and stop values to fit the waveform to the display, and also enables Auto horizontal scaling.

DISPlay:WLAN:EVM:TIME:AUTO:STATe

When enabled, sets the horizontal symbol scale value automatically for the WLAN EVM display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:EVM:TIME:AUTO:STATe {0|1|OFF|ON}
DISPlay:WLAN:EVM:TIME:AUTO:STATe?
```

Arguments `ON or 1` sets the horizontal symbol scale value automatically.

`OFF or 0` specifies to stop setting the horizontal symbol scale value automatically.

Examples `DISP:WLAN:EVM:TIME:AUTO:STATE ON` specifies to the horizontal symbol scale value automatically.

DISPlay:WLAN:EVM:TIME:RESet (No Query Form)

Resets the horizontal symbol start and stop values for the WLAN EVM display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax `DISPlay:WLAN:EVM:TIME:RESet`

Arguments None

Examples `DISP:WLAN:EVM:TIME:RES` resets the horizontal symbol start and stop values for the WLAN EVM display.

DISPlay:WLAN:EVM:TIME:STARt

Sets or queries the horizontal symbol start value for the WLAN EVM display. The units are either Symbols or Seconds, and are set using the command [SENSe]:WLAN:UNIT:TIME.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:EVM:TIME:STARt <NRf>
DISPlay:WLAN:EVM:TIME:STARt?
```

Arguments Floating point number.

Examples `DISP:WLAN:EVM:TIME:START 40` sets the horizontal symbol start value of the WLAN EVM display to 40 Symbols (assuming the units have been set to Symbols).

DISPlay:WLAN:EVM:TIME:STOP

Sets or queries the horizontal symbol stop value for the WLAN EVM display. The units are either Symbols or Seconds, and are set using the command [SENSe]:WLAN:UNIT:TIME.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax

```
DISPlay:WLAN:EVM:TIME:STOP <NRf>
DISPlay:WLAN:EVM:TIME:STOP?
```

Arguments

Floating point number.

Examples

`DISP:WLAN:EVM:TIME:STOP 804.96e-6` sets the horizontal symbol stop value for the WLAN EVM display to 804.960 µs (assuming the units have been set to Seconds).

DISPlay:WLAN:EVM:WINDow:SELect:PLOT

Specifies or queries which graphs are displayed in the WLAN EVM display. Select from Symbols, Subcarriers, or Both.

To set the Time units, use [SENSe]:WLAN:UNIT:TIME. To set the Frequency units, use [SENSe]:WLAN:UNIT:FREQuency. To set the Radix, use [SENSe]:WLAN:RADix.

NOTE. *Use this command only with multicarrier standards and not singlecarrier (802.11b).*

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:EVM:WINDow:SELect:PLOT {SYMBols|SCARier|BOTH}
DISPlay:WLAN:EVM:WINDow:SELect:PLOT?
```

Arguments

`SYMBols` displays the Symbols graph.

`SCARier` displays the Subcarriers graph.

`BOTH` displays both graphs.

Examples

`DISP:WLAN:EVM:WIND:SEL:PLOT SCAR` displays the Subcarrier graph on the WLAN EVM display.

DISPlay:WLAN:EVM:WINDow:TRACe:GRATicule:GRID:STATe

Shows or hides the graticule grid on the screen.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax

```
DISPlay:WLAN:EVM:WINDow:TRACe:GRATicule:GRID:STATe
{0|1|OFF|ON}
DISPlay:WLAN:EVM:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `ON or 1` specifies to show the graticule.

`OFF or 0` specifies to turn it off.

Examples `DISP:WLAN:EVM:WIND:TRAC:GRAT:GRID:STATE ON` specifies to turn the graticule on.

DISPlay:WLAN:EVM:Y:AUTO (No Query Form)

Automatically sets the vertical scale and position values for the WLAN EVM display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax `DISPlay:WLAN:EVM:Y:AUTO`

Arguments None

Examples `DISP:WLAN:EVM:Y:AUTO` automatically sets the vertical scale and position values for the WLAN EVM display.

DISPlay:WLAN:EVM:Y:OFFSet

Sets or queries the value of the offset from the center (vertical position), in percent terms, for the WLAN EVM display. To set the vertical scale, use the command DISPlay:WLAN:EVM:Y[:SCALe].

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax

```
DISPlay:WLAN:EVM:Y:OFFSet <NRf>
DISPlay:WLAN:EVM:Y:OFFSet?
```

Arguments Floating point number between limits which are set as a function of the current scale value guaranteed to keep the EVM plot at least partially visible on the display. Positive inputs push the plot down (as viewed by the user) and negative inputs push the plot up. 0 re-centers the plot.

Returns The value of the offset from the center of the Y axis for the Error Magnitude Vector plot as a floating point number.

Examples `DISP:WLAN:EVM:Y:OFFS 150` sets the offset from the center (vertical position) of the EVM plot to 150 %.

DISPlay:WLAN:EVM:Y:RESet (No Query Form)

Resets the vertical scale and position (offset) values for the WLAN EVM display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax

```
DISPlay:WLAN:EVM:Y:RESet
```

Arguments None

Examples `DISP:WLAN:EVM:Y:RES` specifies to reset the vertical scale and position (offset) values for the WLAN EVM display.

DISPlay:WLAN:EVM:Y[:SCALe]

Sets or queries the vertical scale value for the WLAN EVM display. To set the vertical position (offset) value, use the command DISPlay:WLAN:EVM:Y:OFFSet.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:EVM:Y[:SCALe] <NRf>
DISPlay:WLAN:EVM:Y[:SCALe]?
```

Arguments Floating point number that specifies the vertical scale value for the WLAN EVM display.

Examples `DISP:WLAN:EVM:Y:SCALE 10` sets the vertical scale value for the WLAN EVM display to 10.

DISPlay:WLAN:FLATness:AUTO (No Query Form)

Automatically rescales the horizontal and vertical axes for the best display in the WLAN Spectral Flatness view.

NOTE. *WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax `DISPlay:WLAN:FLATness:AUTO`

Arguments None

Examples `DISP:WLAN:FLAT:AUTO` automatically rescales the horizontal and vertical axes for the best display.

DISPlay:WLAN:FLATness:X:AUTO (No Query Form)

Rescales the horizontal axis automatically in the WLAN Spectral Flatness display.

NOTE. *WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax `DISPlay:WLAN:FLATness:X:AUTO`

Arguments None

Examples `DISP:WLAN:FLAT:X:AUTO` rescales the horizontal axis automatically for the best display.

DISPlay:WLAN:FLATness:X:OFFSet

Sets or queries the value of the offset of the horizontal axis from the center of the WLAN Spectral Flatness display.

NOTE. *WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax

```
DISPlay:WLAN:FLATness:X:OFFSet <NRf>
DISPlay:WLAN:FLATness:X:OFFSet?
```

Arguments Floating point number between limits which are set as a function of the current scale value guaranteed to keep the plot at least partially visible on the display. Positive inputs push the plot down (as viewed by the user) and negative inputs push the plot up. 0 recenters the plot.

Returns The value of the offset of the X axis from the center of the plot as a floating point number.

Examples `DISP:WLAN:FLAT:X:OFFS 150` shifts the display to the left by 150 %.

DISPlay:WLAN:FLATness:X[:SCALe]

Sets or queries the horizontal scale value in the WLAN Spectral Flatness display. The units are Frequency or Subcarrier. To set the units, use [SENSe]:WLAN:UNIT:FREQuency

NOTE. *WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax

```
DISPlay:WLAN:FLATness:X[:SCALe] <NRf>
DISPlay:WLAN:FLATness:X[:SCALe]?
```

Arguments Floating point number that represents the horizontal scale.

Examples `DISP:WLAN:FLAT:X:SCALE 32` sets the horizontal scale value to 32.

DISPlay:WLAN:FLATness:Y:AUTO (No Query Form)

Automatically rescales the vertical axis scale and position values in the WLAN Spectral Flatness display.

NOTE. *WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax `DISPlay:WLAN:FLATness:Y:AUTO`

Arguments None

Examples `DISP:WLAN:FLAT:Y:AUTO` automatically selects the vertical scale and position values.

DISPlay:WLAN:FLATness:Y:OFFSet

Sets or queries the value of the offset from the center (vertical position), in percent terms, for the WLAN Spectral Flatness display. To set the Vertical Scale, use the command DISPlay:WLAN:FLATness:Y[:SCALe].

NOTE. *WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:FLATness:Y:OFFSet <NRf>
DISPlay:WLAN:FLATness:Y:OFFSet?
```

Arguments Floating point number between limits which are set as a function of the current scale value guaranteed to keep the plot at least partially visible on the display. Positive inputs push the plot down (as viewed by the user) and negative inputs push the plot up. 0 re-centers the plot.

Returns The value of the offset from the center of the Y axis for the plot as a floating point number.

Examples `DISP:WLAN:FLAT:Y:OFFS 150` sets the offset from the center (vertical position) of the plot to 150 %.

DISPlay:WLAN:FLATness:Y[:SCALe]

Sets or queries the vertical scale value for the WLAN Spectral Flatness display. To set the vertical position (offset) value, use the command DISPlay:WLAN:FLATness:Y:OFFSet.

NOTE. *WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:FLATness:Y[:SCALe] <NRf>
DISPlay:WLAN:FLATness:Y[:SCALe]?
```

Arguments Floating point number that specifies the vertical scale value.

Examples `DISP:WLAN:FLAT:Y:SCALE 10` sets the vertical scale value to 10.

DISPlay:WLAN:MEASview:DELete (No Query Form)

Deletes the selected measurement view in the WLAN display.

If you attempt to delete a view that is currently displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax

```
DISPlay:WLAN:MEASview:DELete
{SUMMary|FLATness|CONStellation|EVM|MERRor|PERRor|CRESponse|STABle|PVTi
```

Arguments

The following table lists the arguments.

Argument	WLAN measurement view
SUMMary	Summary measurements
FLATness	Spectral Flatness measurement
CONStellation	Constellation measurement
EVM	Error Vector Magnitude measurement
MERRor	Magnitude Error vs. Time
PERRor	Phase Error vs. Time
CRESponse	Channel Response measurement
STABle	Symbol Table
PVTime	Power Vs. Time

Examples

`DISP:WLAN:MEAS:DEL PERR` deletes the Phase Error measurement view.

DISPlay:WLAN:MEASview:NEW (No Query Form)

Displays a new measurement view in the WLAN display.

If you attempt to open a view that is currently displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:MEASview:NEW
{SUMMary|FLATness|CONStellation|EVM|MERRor|PERRor|CRESponse|STABle|PVTi
```

Arguments

The following table lists the arguments.

Argument	WLAN measurement view
SUMMary	Summary measurements
FLATness	Spectral Flatness measurement
CONStellation	Constellation measurement

Argument	WLAN measurement view
EVM	Error Vector Magnitude measurement
MERRor	Magnitude Error vs. Time
PERRor	Phase Error vs. Time
CRESponse	Channel Response measurement
STABle	Symbol Table
PVTime	Power Vs. Time

Examples `DISP:WLAN:MEASVIEW:NEW CONSTELLATION` creates the Constellation measurement view.

DISPlay:WLAN:MEASview:SELect

Selects a measurement view on the screen in the WLAN measurements. The query command returns the currently selected view.

If you attempt to select a view that is not displayed on screen, the error (-200, "Execution error; Measurement is already running") will be returned.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax

```
DISPlay:WLAN:MEASview:SELect
{SUMMary|FLATness|CONStellation|EVM|MERRor|PERRor|CRESponse|STABle|P
DISPlay:WLAN:MEASview:SELect?
```

Arguments The following table lists the arguments.

Argument	WLAN measurement view
SUMMary	Summary measurements
FLATness	Spectral Flatness measurement
CONStellation	Constellation measurement
EVM	Error Vector Magnitude measurement
MERRor	Magnitude Error vs. Time
PERRor	Phase Error vs. Time
CRESponse	Channel Response measurement
STABle	Symbol Table
PVTime	Power Vs. Time

Examples

`DISP:WLAN:MEASVIEW:SEL CONSTELLATION` selects the Constellation measurement view.

`DISP:WLAN:MEASVIEW:SEL?` might return FLAT, indiciating that the Spectral Flatness measurement view is the currently selected view.

DISPlay:WLAN:MERRor:FREQuency:AUTO (No Query Form)

Automatically rescales the horizontal subcarrier (frequency) start and stop values to fit the waveform to the screen in the WLAN Magnitude Error display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

`DISPlay:WLAN:MERRor:FREQuency:AUTO`

Arguments

None

Examples

`DISPlay:WLAN:MERRor:FREQuency:AUTO` automatically rescales the horizontal Subcarrier (frequency) values to fit the waveform to the screen.

DISPlay:WLAN:MERRor:FREQuency:RESet (No Query Form)

Automatically resets the horizontal subcarrier (frequency) start and stop values for the WLAN Magnitude Error display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

`DISPlay:WLAN:MERRor:FREQuency:RESet`

Arguments

None

Examples `DISP:WLAN:MERR:FREQ:RES` automatically resets the horizontal subcarrier start and stop values for the WLAN Magnitude Error display.

DISPlay:WLAN:MERRor:FREQuency:STARt

Specifies or queries the horizontal start value (left edge) on the WLAN Magnitude Error display. The units are either Frequency (Hz) or Subcarriers, and are set using the command [SENSe]:WLAN:UNIT:FREQuency.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax

```
DISPlay:WLAN:MERRor:FREQuency:STARt <NRf>
DISPlay:WLAN:MERRor:FREQuency:STARt?
```

Arguments Floating point number that specifies the horizontal start value (left edge) on the graph.

Returns The returned value indicates the Subcarrier (for non-b standards only) or frequency shown at the left edge of the display.

Examples `DISP:WLAN:MERR:FREQ:START 2.41e9` sets the horizontal start value (left edge) to 2.41e9 MHz (assuming the units have been set to Frequency).

DISPlay:WLAN:MERRor:FREQuency:STOP

Specifies or queries the horizontal stop value (right edge) on the WLAN Magnitude Error display. The units are either Frequency (Hz) or Subcarriers, and are set using the command [SENSe]:WLAN:UNIT:FREQuency.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax

```
DISPlay:WLAN:MERRor:FREQuency:STOP <NRf>
DISPlay:WLAN:MERRor:FREQuency:STOP?
```

Arguments

Floating point number that specifies the horizontal stop value (right edge) on the display.

Examples

`DISP:WLAN:MERR:FREQ:STOP 8.1250` sets the horizontal stop value to 8.1250 MHz (assuming the units have been set to Frequency).

DISPlay:WLAN:MERRor:MARKer:SHOW:STATe

Shows or hides the marker readout on the WLAN Magnitude Error display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:MERRor:MARKer:SHOW:STATe {0|1|OFF|ON}
DISPlay:WLAN:MERRor:MARKer:SHOW:STATe?
```

Arguments

`ON or 1` specifies to show the marker readout.

`OFF or 0` specifies to turn the marker readout off.

Examples

`DISP:WLAN:MERR:MARK:SHOW:STATE ON` specifies to turn the marker readout on .

DISPlay:WLAN:MERRor[:SCALe]:AUTO (No Query Form)

Automatically adjusts all vertical and horizontal values to provide the best display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax `DISPlay:WLAN:MERRor[:SCALe]:AUTO`

Arguments None

Examples `DISP:WLAN:MERR:SCALE:AUTO` automatically adjusts all vertical and horizontal values to provide the best display.

DISPlay:WLAN:MERRor:TIME:AUTO (No Query Form)

Sets the horizontal start and stop values to fit the waveform to the WLAN Magnitude Error display, and also enables Auto horizontal scaling so the waveform will remain properly scaled as burst lengths change.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax `DISPlay:WLAN:MERRor:TIME:AUTO`

Arguments None

Examples `DISP:WLAN:MERR:TIME:AUTO` sets the horizontal start and stop values to fit the waveform to the display, and also enables Auto horizontal scaling.

DISPlay:WLAN:MERRor:TIME:AUTO:STATe

When enabled, automatically sets the horizontal symbol scale value for the WLAN MERR display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax

```
DISPlay:WLAN:MERRor:TIME:AUTO:STATe {0|1|OFF|ON}
DISPlay:WLAN:MERRor:TIME:AUTO:STATe?
```

Arguments

`ON or 1` sets the horizontal symbol scale value automatically.

`OFF or 0` specifies to stop setting the horizontal symbol scale value automatically.

Examples

`DISP:WLAN:MERR:TIME:AUTO:STATE ON` specifies to the horizontal symbol scale value automatically.

DISPlay:WLAN:MERRor:TIME:RESet (No Query Form)

Resets the horizontal symbol start and stop values for the WLAN Magnitude Error display.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:MERRor:TIME:RESet
```

Arguments

None

Examples

`DISP:WLAN:MERR:TIME:RES` resets the horizontal symbol start and stop values for the WLAN Magnitude Error display.

DISPlay:WLAN:MERRor:TIME:STARt

Sets or queries the horizontal symbol start value for the WLAN Magnitude Error display. The units are either Symbols or Seconds, and are set using the command [SENSe]:WLAN:UNIT:TIME.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:MERRor:TIME:STARt <NRf>
DISPlay:WLAN:MERRor:TIME:STARt?
```

Arguments Floating point number.

Examples `DISP:WLAN:MERR:TIME:START 40` sets the horizontal symbol start value of the WLAN Magnitude Error display to 40 Symbols (assuming the units have been set to Symbols).

DISPlay:WLAN:MERRor:TIME:STOP

Sets or queries the horizontal symbol stop value for the WLAN Magnitude Error display. The units are either Symbols or Seconds, and are set using the command [SENSe]:WLAN:UNIT:TIME.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax

```
DISPlay:WLAN:MERRor:TIME:STOP <NRf>
DISPlay:WLAN:MERRor:TIME:STOP?
```

Arguments Floating point number.

Examples `DISP:WLAN:MERR:TIME:STOP 804.96e-6` sets the horizontal symbol stop value for the WLAN Magnitude Error display to 804.960 µs (assuming the units have been set to Seconds).

DISPlay:WLAN:MERRor:WINDow:SELect:PLOT

Specifies or queries which graphs are displayed in the WLAN Magnitude Error display. Select from Symbols, Subcarriers, or Both.

To set the Time units, use [SENSe]:WLAN:UNIT:TIME. To set the Frequency units, use [SENSe]:WLAN:UNIT:FREQuency. To set the Radix, use [SENSe]:WLAN:RADix.

NOTE. *Use this command only with multicarrier standards and not singlecarrier (802.11b).*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:MERRor:WINDow:SELect:PLOT
{SYMBols|SCARier|BOTH}
DISPlay:WLAN:MERRor:WINDow:SELect:PLOT?
```

Arguments `SYMBols` displays the Symbols graph.

`SCARier` displays the Subcarriers graph.

`BOTH` displays both graphs.

Examples `DISP:WLAN:MERR:WIND:SEL:PLOT SCAR` displays the Subcarrier graph on the WLAN Magnitude Error display.

DISPlay:WLAN:MERRor:WINDow:TRACe:GRATicule:GRID:STATe

Shows or hides the graticule grid on the display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:MERRor:WINDow:TRACe:GRATicule:GRID:STATe
{0|1|OFF|ON}
DISPlay:WLAN:MERRor:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `ON or 1` specifies to show the graticule.

`OFF or 0` specifies to turn it off.

Examples `DISP:WLAN:MERR:WIND:TRAC:GRAT:GRID:STATE ON` specifies to turn the graticule on.

DISPlay:WLAN:MERRor:Y:AUTO (No Query Form)

Automatically sets the vertical scale and position values for the WLAN Magnitude Error display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax `DISPlay:WLAN:MERRor:Y:AUTO`

Arguments None

Examples `DISP:WLAN:MERR:Y:AUTO` automatically sets the vertical scale and position values for the WLAN Magnitude Error display.

DISPlay:WLAN:MERRor:Y:OFFSet

Sets or queries the offset from the center (vertical position), in percent terms, for the WLAN Magnitude Error display. To set the vertical scale, use the command DISPlay:WLAN:EVM:Y[:SCALe].

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:MERRor:Y:OFFSet <NRf>
DISPlay:WLAN:MERRor:Y:OFFSet?
```

Arguments Floating point number between limits which are set as a function of the current scale value guaranteed to keep the Magnitude Error plot at least partially visible

on the display. Positive inputs push the plot down (as viewed by the user) and negative inputs push the plot up. 0 re-centers the plot.

Returns The offset from the center of the Y axis for the Magnitude Error plot as a floating point number.

Examples `DISP:WLAN:MERR:Y:OFFS 150` sets the offset from the center of the vertical axis of the Magnitude Error plot to 150 %.

DISPlay:WLAN:MERRor:Y:RESet (No Query Form)

Resets the vertical scale and position (offset) for the WLAN Magnitude Error display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax `DISPlay:WLAN:MERRor:Y:RESet`

Arguments None

Examples `DISP:WLAN:MERR:Y:RES` specifies to reset the vertical scale and position (offset) values for the WLAN Magnitude Error display.

DISPlay:WLAN:MERRor:Y[:SCALe]

Sets or queries the vertical scale value for the WLAN Magnitude Error display. To set the vertical position (offset) value, use the command DISPlay:WLAN:EVM:Y:OFFSet.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax

```
DISPlay:WLAN:MERRor:Y[:SCALe] <NRf>
DISPlay:WLAN:MERRor:Y[:SCALe]?
```

Arguments Floating point number that specifies the vertical scale value for the WLAN Magnitude Error display.

Examples `DISP:WLAN:MERR:Y:SCALE 10` sets the vertical scale for the WLAN Magnitude Error display to 10.

DISPlay:WLAN:PERRor:FREQuency:AUTO (No Query Form)

Automatically rescales the horizontal subcarrier (frequency) start and stop values to fit the waveform to the screen in the WLAN Phase Error display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax

```
DISPlay:WLAN:PERRor:FREQuency:AUTO
```

Arguments None

Examples `DISP:WLAN:PERR:FREQ:AUTO` automatically rescales the horizontal Subcarrier (frequency) start and stop values to fit the waveform to the screen.

DISPlay:WLAN:PERRor:FREQuency:RESet (No Query Form)

Automatically resets the horizontal subcarrier (frequency) start and stop values for the WLAN Phase Error display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax `DISPlay:WLAN:PERRor:FREQuency:RESet`

Arguments None

Examples `DISP:WLAN:PERR:FREQ:RES` automatically resets the horizontal Subcarrier start and stop values for the WLAN Phase Error display.

DISPlay:WLAN:PERRor:FREQuency:STARt

Specifies or queries the horizontal start value (left edge) on the WLAN Phase Error display. The units are either Frequency (Hz) or Subcarriers, and are set using the command [SENSe]:WLAN:UNIT:FREQuency.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:PERRor:FREQuency:STARt <NRf>
DISPlay:WLAN:PERRor:FREQuency:STARt?
```

Arguments Floating point number that specifies the horizontal start value (left edge) on the display.

Returns The readout indicates the Subcarrier (for non-b standards only) or frequency shown at the left edge of the display.

Examples `DISP:WLAN:PERR:FREQ:START -8.1250` sets the horizontal start value to -8.1250 MHz (assuming the units have been set to Frequency).

DISPlay:WLAN:PERRor:FREQuency:STOP

Specifies or queries the horizontal stop value (right edge) on the WLAN Phase Error display. The units are either Frequency (Hz) or Subcarriers, and are set using the command [SENSe]:WLAN:UNIT:FREQuency.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:PERRor:FREQuency:STOP <NRf>
DISPlay:WLAN:PERRor:FREQuency:STOP?
```

Arguments Floating point number that specifies the horizontal stop value (right edge) on the display.

Examples `DISP:WLAN:PERR:FREQ:STOP 8.1250` sets the horizontal stop value to 8.1250 MHz (assuming the units have been set to Frequency).

DISPlay:WLAN:PERRor:MARKer:SHOW:STATe

Show or hides the Marker readout on the WLAN Phase Error display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:PERRor:MARKer:SHOW:STATe {0|1|OFF|ON}
DISPlay:WLAN:PERRor:MARKer:SHOW:STATe?
```

Arguments `ON or 1` specifies to show the Marker readout.

`OFF or 0` specifies to turn the Marker readout off.

Examples `DISP:WLAN:PERR:MARK:SHOW:STATE ON` specifies to turn the Marker readout on.

DISPlay:WLAN:PERRor[:SCALe]:AUTO (No Query Form)

Automatically adjusts all vertical and horizontal settings to provide the best display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax `DISPlay:WLAN:PERRor[:SCALe]:AUTO`

Arguments None

Examples `DISP:WLAN:PERRor:SCALE:AUTO` automatically adjusts all vertical and horizontal settings to provide the best display.

DISPlay:WLAN:PERRor:TIME:AUTO (No Query Form)

Automatically sets the horizontal symbol start and stop values to fit the waveform to the screen for the WLAN Phase Error vs. Symbol plot. In addition, sets the state for this feature to ON.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax `DISPlay:WLAN:PERRor:TIME:AUTO`

Arguments None

Examples `DISP:WLAN:PERR:TIME:AUTO` sets the instrument to automatically select the horizontal symbol start and stop values to fit the waveform to the screen, and turns the AUTO feature on.

DISPlay:WLAN:PERRor:TIME:AUTO:STATe

When enabled, automatically sets the horizontal symbol scale value for the WLAN Phase Error display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:PERRor:TIME:AUTO:STATe {0|1|OFF|ON}
DISPlay:WLAN:PERRor:TIME:AUTO:STATe?
```

Arguments `ON or 1` specifies to set the Analysis Time value automatically.

`OFF or 0` specifies to turn stop setting the Analysis Time value automatically.

Examples `DISP:WLAN:PERR:TIME:AUTO:STATE ON` specifies to set the Analysis Time value automatically, based on the requirements of the selected display

DISPlay:WLAN:PERRor:TIME:RESet (No Query Form)

Resets the horizontal symbol start and stop values for the WLAN Phase Error Average vs. Symbol display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax `DISPlay:WLAN:PERRor:TIME:RESet`

Arguments None

Examples `DISP:WLAN:PERR:TIME:RES` resets the horizontal symbol start and stop values for the WLAN Phase Error Average vs. Symbol display.

DISPlay:WLAN:PERRor:TIME:STARt

Sets or queries the horizontal symbol start value for the WLAN Phase Error Average vs. Symbol display. The units are either Symbols or Seconds, and are set using the command [SENSe]:WLAN:UNIT:TIME.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:PERRor:TIME:STARt <NRf>
DISPlay:WLAN:PERRor:TIME:STARt?
```

Arguments Floating point number.

Examples `DISP:WLAN:PERR:TIME:START 40` sets the horizontal symbol start value of the Average vs. Symbol display to 40 Symbols (assuming the units have been set to Symbols).

DISPlay:WLAN:PERRor:TIME:STOP

Sets or queries the horizontal symbol stop value for the WLAN Phase Error Average vs. Symbol display. The units are either Symbols or Seconds, and are set using the command [SENSe]:WLAN:UNIT:TIME.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:PERRor:TIME:STOP <NRf>
DISPlay:WLAN:PERRor:TIME:STOP?
```

Arguments Floating point number.

Examples `DISP:WLAN:PERR:TIME:STOP 804.960` sets the horizontal symbol stop value for the Average vs. Symbol display to 804.960 µs(assuming the units have been set to Seconds).

DISPlay:WLAN:PERRor:WINDow:SELect:PLOT

Specifies or queries which displays are displayed in the WLAN Phase Error analysis display (Symbols, Subcarriers, or Both).

To set the Time units, use [SENSe]:WLAN:UNIT:TIME. To set the Frequency units, use [SENSe]:WLAN:UNIT:FREQuency. To set the Radix, use [SENSe]:WLAN:RADix.

NOTE. *Use this command only with multicarrier standards and not singlecarrier (802.11b).*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax

```
DISPlay:WLAN:PERRor:WINDow:SELect:PLOT
{SYMBols|SCARier|BOTH}
DISPlay:WLAN:PERRor:WINDow:SELect:PLOT?
```

Arguments `SYMBols` displays the Symbols graph.

`SCARier` displays the Subcarriers graph.

`BOTH` displays both graphs.

Examples `DISP:WLAN:PERR:WIND:SEL:PLOT SCAR` displays the Subcarrier graph on the WLAN Phase Error analysis display.

DISPlay:WLAN:PERRor:WINDow:TRACe:GRATicule:GRID:STATe

Shows or hides the graticule grid on the screen.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax

```
DISPlay:WLAN:PERRor:WINDow:TRACe:GRATicule:GRID:STATe
{0|1|OFF|ON}
DISPlay:WLAN:PERRor:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments

`ON or 1` specifies to show the graticule.

`OFF or 0` specifies to turn it off.

Examples

`DISP:WLAN:PERR:WIND:TRAC:GRAT:GRID:STATE ON` specifies to turn the graticule on.

DISPlay:WLAN:PERRor:Y:AUTO (No Query Form)

Automatically selects the vertical scale and position values for the WLAN Phase Error Symbol and Subcarrier plots.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PERRor:Y:AUTO
```

Arguments

None

Examples

`DISP:WLAN:PERR:Y:AUTO` automatically selects the vertical scale and position values for the WLAN Phase Error Symbol and Subcarrier plots.

DISPlay:WLAN:PERRor:Y:OFFSet

Sets or queries the value of the offset from the center (vertical position), in percent terms, for the WLAN Phase Error Symbol and Subcarrier plots. To set the Vertical Scale, use the command DISPlay:WLAN:PERRor:Y[:SCALe].

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PERRor:Y:OFFSet <NRf>
DISPlay:WLAN:PERRor:Y:OFFSet?
```

Arguments

Floating point number between limits which are set as a function of the current scale value guaranteed to keep the plot at least partially visible on the display. Positive inputs push the plot down (as viewed by the user) and negative inputs push the plot up. 0 re-centers the plot.

Returns

The value of the offset from the center of the Y axis for the Phase Error plot as a floating point number.

Examples

`DISP:WLAN:PERR:Y:OFFS 150` sets the offset from the center (vertical position) of the Phase Error plot to 150 %.

DISPlay:WLAN:PERRor:Y:RESet (No Query Form)

Resets the vertical scale and position (offset) values for the WLAN Phase Error Symbol and Subcarrier plots.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Display commands

Syntax

```
DISPlay:WLAN:PERRor:Y:RESet
```

Arguments

None

Examples

`DISP:WLAN:PERR:Y:RES` specifies to reset the vertical scale and position (offset) values for the WLAN Phase Error plot.

DISPlay:WLAN:PERRor:Y[:SCALe]

Sets or queries the vertical scale value for the WLAN Phase Error plots. To set the vertical position (offset) value, use the command DISPlay:WLAN:PERRor:Y:OFFSet.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:PERRor:Y[:SCALe] <NRf>
DISPlay:WLAN:PERRor:Y[:SCALe]?
```

Arguments Floating point number that specifies the vertical scale value.

Examples `DISP:WLAN:PERR:Y:SCALE 10` sets the vertical scale value to 10.

DISPlay:WLAN:PVTime:BURSt:X[:SCALe]

Sets or queries the value of the scale (width), in seconds, for the WLAN Power vs. Time display when using horizontal Full Burst view.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:PVTime:BURSt:X[:SCALe] <NRf>
DISPlay:WLAN:PVTime:BURSt:X[:SCALe]?
```

Arguments Floating point number that represents the value of the scale (width), in seconds, when using horizontal Full Burst view.

Examples `DISP:WLAN:PVT:BURS:X:SCALE 5.0E-6` sets the width of the display to 5.000 µS for the horizontal Full Burst view.

DISPlay:WLAN:PVTime:BURSt:X[:SCALe]:AUTO (No Query Form)

Automatically sets the starting time (position) and scale (width) values for the best display in the horizontal Full Burst view.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax `DISPlay:WLAN:PVTime:BURSt:X[:SCALe]:AUTO`

Arguments None

Examples `DISP:WLAN:PVT:BURS:X:SCALE:AUTO` automatically sets the starting time (position) and scale (width) values for the best display.

DISPlay:WLAN:PVTime:BURSt:X[:SCALe]:OFFSet

Sets or queries the starting time (position) value, in seconds, for the WLAN Power vs. Time display when using the horizontal Full Burst view.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:PVTime:BURSt:X[:SCALe]:OFFSet <NRf>
DISPlay:WLAN:PVTime:BURSt:X[:SCALe]:OFFSet?
```

Arguments Floating point number that represents the starting time (position) value, in seconds, when using horizontal Full Burst view.

Examples `DISP:WLAN:PVT:BURS:X:OFFFSET 5.0E-6` sets the starting time (position) of the graph to 5.000 µS for horizontal Full Burst view.

DISPlay:WLAN:PVTime:BURSt:X[:SCALe]:RESet (No Query Form)

Resets the starting time (position) and scale (width) values for the horizontal Full Burst view on the WLAN Power vs. Time display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax `DISPlay:WLAN:PVTime:BURSt:X[:SCALe]:RESet`

Arguments None

Examples `DISP:WLAN:PVT:BURST:X:SCALE:RESET` resets the starting time (position) and scale (width) values for the horizontal Full Burst view.

DISPlay:WLAN:PVTime:FALL:X[:SCALe]

Sets or queries the scale (width) value, in seconds, for the WLAN Power vs. Time display when using the horizontal Falling Edge view.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:PVTime:FALL:X[:SCALe] <NRf>
DISPlay:WLAN:PVTime:FALL:X[:SCALe]?
```

Arguments Floating point number that represents the scale (width), in seconds, when using the horizontal Falling Edge view.

Examples `DISP:WLAN:PVT:FALL:X:SCALE 5.0E-6` sets the width of the graph to 5.000 µS when using the horizontal Falling Edge view.

DISPlay:WLAN:PVTime:FALL:X[:SCALe]:AUTO (No Query Form)

Automatically sets the starting time (position) and scale (width) values for the best display in the horizontal Falling Edge view.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax `DISPlay:WLAN:PVTime:FALL:X[:SCALe]:AUTO`

Arguments None

Examples `DISP:WLAN:PVT:FALL:X:SCALE:AUTO` automatically sets the starting time (position) and scale (width) values for the best display.

DISPlay:WLAN:PVTime:FALL:X[:SCALe]:OFFSet

Sets or queries the value for the starting time (offset) of the graph, in seconds, for the WLAN Power vs. Time display when using the horizontal Falling Edge view.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:PVTime:FALL:X[:SCALe]:OFFSet <NRf>
DISPlay:WLAN:PVTime:FALL:X[:SCALe]:OFFSet?
```

Arguments Floating point number that represents the value of the starting time (offset) of the graph, in seconds, when using the horizontal Falling Edge view.

Examples `DISP:WLAN:PVT:FALL:X:SCALE 5.0` sets the starting time (position) of the graph to 5.000 µS when using the horizontal Falling Edge view.

DISPlay:WLAN:PVTime:FALL:X[:SCALe]:RESet (No Query Form)

Resets the starting time (position) and scale (width) values for the horizontal Falling Edge view on the WLAN Power vs. Time display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax `DISPlay:WLAN:PVTime:FALL:X[:SCALe]:RESet`

Arguments None

Examples `DISP:WLAN:PVT:FALL:X:SCALE:RESET` resets the starting time (position) and scale (width) values for the horizontal Falling Edge view.

DISPlay:WLAN:PVTime:MARKer:SHOW:STATe

Shows or hides the marker readout on the WLAN Power vs. Time display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:PVTime:MARKer:SHOW:STATe {0|1|OFF|ON}
DISPlay:WLAN:PVTime:MARKer:SHOW:STATe?
```

Arguments `ON or 1` specifies to show the marker readout.

`OFF or 0` specifies to turn the marker readout off.

Examples `DISP:WLAN:PVT:MARK:SHOW:STATE ON` specifies to turn the marker readout on.

DISPlay:WLAN:PVTime:RISE:X[:SCALe]

Sets or queries the scale (width) value, in seconds, for the WLAN Power vs. Time display when using the horizontal Rising Edge view.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:PVTime:RISE:X[:SCALe] <NRf>
DISPlay:WLAN:PVTime:RISE:X[:SCALe]?
```

Arguments Floating point number that represents the scale (width) value, in seconds, for the horizontal Rising Edge view.

Examples `DISP:WLAN:PVT:RISE:X:SCALE 5.0E-6` sets the width of the graph to 5.000 µS for the horizontal Rising Edge view.

DISPlay:WLAN:PVTime:RISE:X[:SCALe]:AUTO (No Query Form)

Automatically sets the starting time (position) and scale (width) values for the best display in the horizontal Rising Edge view of the WLAN Power vs. Time display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:PVTime:RISE:X[:SCALe]:AUTO
```

Arguments None

Examples `DISP:WLAN:PVT:RISE:X:SCALE:AUTO` automatically sets the starting time (position) and scale (width) values for the best display.

DISPlay:WLAN:PVTime:RISE:X[:SCALe]:OFFSet

Sets or queries the starting time value (offset), in seconds, for the WLAN Power vs. Time display when using the horizontal Rising Edge view.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:PVTime:RISE:X[:SCALe]:OFFSet <NRf>
DISPlay:WLAN:PVTime:RISE:X[:SCALe]:OFFSet?
```

Arguments Floating point number that represents the value of the starting time (offset) of the graph, in seconds, for the horizontal Rising Edge view.

Examples `DISP:WLAN:PVT:RISE:X:OFFSET 5.0e-6` sets the starting time (position) of the graph to 5.000 µS for the horizontal Rising Edge view.

DISPlay:WLAN:PVTime:RISE:X[:SCALe]:RESet (No Query Form)

Resets the starting time (position) and scale (width) values for the horizontal Rising Edge view on the WLAN Power vs. Time display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:PVTime:RISE:X[:SCALe]:RESet
```

Arguments None

Examples `DISP:WLAN:PVT:RISE:X:SCALE:RESET` resets the starting time (position) and scale (width) values for the horizontal Rising Edge view.

DISPlay:WLAN:PVTime:WINDow:SELect:PLOT

Specifies or queries which view to use for the WLAN Power vs. Time display. Select from Full Burst, Rising Edge or Falling Edge.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:PVTime:WINDow:SELect:PLOT {BURSt|RISE|FALL}
DISPlay:WLAN:PVTime:WINDow:SELect:PLOT?
```

Arguments `BURSt` sets the view to Full Burst, which displays the entire packet, with vertical lines indicating the length of the burst.

`RISE` sets the view to Rising Edge, which zooms the display into the interval around the burst rising edge, with vertical lines indicating the 10% to 90% Power-On Ramp time.

`FALL` sets the view to Falling Edge, which zooms the display into the interval around the burst Falling Edge, with vertical lines indicating the 90% to 10% Power-Down Ramp time.

Returns `DISP:WLAN:PVT:WIND:SEL:PLOT BURS` sets the view to Full Burst.

`DISP:WLAN:PVT:WIND:SEL:PLOT?` might return `RISE`, indicating that the view has been set to Rising Edge.

DISPlay:WLAN:PVTime:WINDow:TRACe:GRATicule:GRID:STATe

Shows or hides the graticule grid on the screen.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:PVTime:WINDow:TRACe:GRATicule:GRID:STATe
{0|1|OFF|ON}
DISPlay:WLAN:PVTime:WINDow:TRACe:GRATicule:GRID:STATe?
```

Arguments `ON or 1` specifies to show the graticule.

`OFF or 0` specifies to turn it off.

Examples `DISP:WLAN:PVT:WIND:TRAC:GRAT:GRID:STATE ON` specifies to turn the graticule on.

DISPlay:WLAN:PVTime:Y[:SCALe]

Sets or queries the vertical scale value for the WLAN Power vs. Time display.

To set the vertical position (offset) value, use the command DISPlay:WLAN:PVTime:Y[:SCALe]:OFFSet.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax

```
DISPlay:WLAN:PVTime:Y[:SCALe] <NRf>
DISPlay:WLAN:PVTime:Y[:SCALe]?
```

Arguments Floating point number that specifies the vertical scale value for the WLAN Power vs. Time display.

Examples `DISP:WLAN:PVT:Y:SCALE 10` sets the vertical scale value to 10 dB.

DISPlay:WLAN:PVTime:Y[:SCALe]:AUTO (No Query Form)

Automatically selects the vertical scale and position values for the WLAN Power vs. Time display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax `DISPlay:WLAN:PVTime:Y[:SCALe]:AUTO`

Arguments None

Examples `DISP:WLAN:PVT:Y:AUTO` automatically selects the vertical scale and position values for the display.

DISPlay:WLAN:PVTime:Y[:SCALe]:OFFSet

Sets or queries the value of the vertical offset (top edge of the vertical axis) for the WLAN Power vs. Time display. To set the vertical scale, use the command DISPlay:WLAN:PVTime:Y[:SCALe].

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:PVTime:Y[:SCALe]:OFFSet <NRf>
DISPlay:WLAN:PVTime:Y[:SCALe]:OFFSet?
```

Arguments Floating point number between limits which specifies the value of the vertical offset. Positive inputs push the plot down (as viewed by the user) and negative inputs push the plot up.

Returns The value of the vertical offset for the WLAN Power vs. Time display as a floating point number.

Examples `DISP:WLAN:PVT:Y:OFFS -10` sets the vertical offset to -10 dBm.

DISPlay:WLAN:PVTime:Y[:SCALe]:PDIVision

Sets or queries the value of the vertical scale for the WLAN Power vs. Time display, in dB/division. This is only a visual control for panning the graph.

Group Display commands

Syntax
```
DISPlay:WLAN:PVTime:Y[:SCALe]:PDIVision <NRf>
DISPlay:WLAN:PVTime:Y[:SCALe]:PDIVision?
```

Arguments Floating point value that represents the value of the vertical scale, in dB/division, for the WLAN Power vs. Time display.

Examples `DISP:WLAN:PVT:Y:SCALE:PDIV 20` sets the vertical scale to 20.0 dB/division.

DISPlay:WLAN:PVTime:Y[:SCALe]:RESet (No Query Form)

Resets the vertical scale and position (offset) values for the WLAN Power vs. Time display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:PVTime:Y[:SCALe]:RESet
```

Arguments None

Examples `DISP:WLAN:PVT:Y:RES` specifies to reset the vertical scale and position (offset) values for the WLAN Power vs. Time display.

DISPlay:WLAN:SUMMary:EVMUnits

Sets or queries the EVM Units setting in the WLAN Summary display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Display commands

Syntax
```
DISPlay:WLAN:SUMMary:EVMUnits {PERCent | DB}
DISPlay:WLAN:SUMMary:EVMUnits?
```

Arguments PERCent. Units in the WLAN Summary display are shown as percentages.

DB. Units in the WLAN Summary display are shown in decibels (dB).

*ESE

Sets or queries the bits in the Event Status Enable Register (ESER). The ESER prevents events from being reported to the Status Byte Register (STB). Refer to Section 3, *Status and Events*, for the register information.

Conditions Measurement views: All

Group IEEE common commands

Syntax
```
*ESE <value>
*ESE?
```

Related Commands *CLS, *ESR?, *SRE, *STB?

Arguments `<value> ::  <NR1>` is a value in the range from 0 through 255. The binary bits of the ESER are set according to this value.

Examples `*ESE 145` sets the ESER to binary 10010001, which enables the PON, EXE, and OPC bits.

`*ESE?` might return the string `*ESE 184`, showing that the ESER contains the binary value 10111000.

*ESR? (Query Only)

Returns the contents of the Standard Event Status Register (SESR). *ESR? also clears the SESR (since reading the SESR clears it). Refer to Section 3, *Status and Events*, for the register information.

Conditions Measurement views: All

Group IEEE common commands

Syntax
```
*ESR?
```

Related Commands *CLS, *ESE, *SRE, *STB?

Arguments None

Returns <NR1> representing the contents of the SESR by a 0 to 255 decimal number.

Examples *ESR? might return the value 213, showing that the SESR contains binary 11010101.

FETCh:ACPower? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the Channel power and ACPR measurement results for all available channels.

Conditions Measurement views: Channel power and ACPR

Group Fetch commands

Syntax FETCh:ACPower?

Arguments None

Returns

```
<chan_power>,<acpr_lower(1)>,<acpr_upper(1)>,
<acpr_lower(2)>,<acpr_upper(2)>,...
<acpr_lower(n)>,<acpr_upper(n)>
```

Where
<chan_power> is the average power of the main channel as the power reference in dBm. The unit can be changed by the [SENSe]:POWer:UNITs command.
<acpr_lower(n)> is the ACPR for the lower channel #n in dB.
<acpr_upper(n)> is the ACPR for the upper channel #n in dB.

The number of n depends on the setting of the [SENSe]:ACPower:CHANnel:PAIRs command.

Examples FETCH:ACPOWER? might return
4.227,-28.420,-23.847,-22.316,-29.225, indicating
(average power of the main channel) = 4.227 dBm,
(ACPR for the lower channel 1) = -28.420 dB,
(ACPR for the upper channel 1) = -23.847 dB,
(ACPR for the lower channel 2) = -22.316 dB, and
(ACPR for the upper channel 2) = -29.225 dB.

FETCh:ACPower:CHANnel:POWer? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the average power of the main channel (power reference) in the Channel power and ACPR measurement.

Conditions Measurement views: Channel power and ACPR

Group Fetch commands

Syntax `FETCh:ACPower:CHANnel:POWer?`

Arguments None

Returns `<chan_power> :: <NRf>` is the average power of the main channel in dBm. The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `FETCH:ACPOWER:CHANNEL:POWER?` might return `4.227`, indicating that the average power of the main channel is 4.227 dBm.

FETCh:ACPower:SPECtrum? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns spectrum trace data of the Channel power and ACPR measurement.

Conditions Measurement views: Channel power and ACPR

Group Fetch commands

Syntax `FETCh:ACPower:SPECtrum?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.
<data(n)> is the amplitude in dBm at the nth data point,
4-byte little endian floating-point format specified in IEEE 488.2.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples FETCH:ACPOWER:SPECTRUM? might return #43204xxxx... (3204-byte data) for the spectrum trace data of the Channel power and ACPR measurement.

FETCh:{AM|FM|PM}? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the trace data in the AM/FM/PM measurement.

Conditions Measurement views: AM, FM, and PM

Group Fetch commands

Syntax FETCh:{AM|FM|PM}?

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the modulation factor in percent (AM), frequency deviation in Hz (FM), or phase deviation in degrees (PM) at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples FETCH:AM? might return #3156xxxx... (156-byte data) for the AM measurement trace.

FETCh:AM:AMINdex? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the modulation depth which is the (positive peak modulation factor - negative peak modulation factor)/2 returned as a percentage (%).

Conditions Measurement views: AM

Group Fetch commands

Syntax `FETCh:AM:AMINdex?`

Arguments None

Returns `<value> ::` `<NRf>` the modulation index.

Examples `FETCH:AM:AMINDEX?` might return `77.1854035556E-3` , indicating the modulation index is 0.0772% or 77.2 m%.

FETCh:AM:AMNegative? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the negative peak modulation factor (–AM) in the AM measurement.

Conditions Measurement views: AM

Group Fetch commands

Syntax `FETCh:AM:AMNegative?`

Arguments None

Returns `<-AM> ::` `<NRf>` is the negative peak modulation factor in percent (%).

Examples `FETCH:AM:AMNEGATIVE?` might return `–23.4`, indicating the negative peak modulation factor is –23.4%.

FETCh:AM:AMPositive? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the positive peak modulation factor (+AM) in the AM measurement.

Conditions Measurement views: AM

Group Fetch commands

Syntax `FETCh:AM:AMPositive?`

Arguments None

Returns `<+AM> :: <NRf>` is the positive peak modulation factor in percent (%).

Examples `FETCH:AM:AMPOSITIVE?` might return `43.8`, indicating the positive peak modulation factor is 43.8%.

FETCh:AM:RESult? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the AM measurement results.

Conditions Measurement views: AM

Group Fetch commands

Syntax `FETCh:AM:RESult?`

Arguments None

Returns `<+AM>,<-AM>,<Total AM>`

Where
`<+AM> :: <NRf>` is the positive peak modulation factor in percent (%).
`<-AM> :: <NRf>` is the negative peak modulation factor in percent (%).
`<Total AM> :: <NRf>` is the (peak to peak modulation factor)/2 in percent (%).

Examples `FETCH:AM:RESULT?` might return `62.63,-50.89,56.76`.

FETCh:AUDio:FERRor? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the carrier frequency error in the audio measurement.

This command is only available for FM and PM measurements.

Conditions Measurement views: Audio Summary

Group Fetch commands

Syntax `FETCh:AUDio:FERRor?`

Arguments None

Returns <value> ::= <NRf> the carrier frequency error.

Examples `FETCH:AUDIO:FERROR?` might return `419.9529809622` indicating that the audio frequency error is approximately 420 Hz.

FETCh:AUDio:FREQuency? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the audio frequency.

Conditions Measurement views: Audio Summary

Group Fetch commands

Syntax `FETCh:AUDio:FREQuency?`

Arguments None

Returns <value> ::= <NRf> the audio frequency.

Examples `FETCH:AUDIO:FREQUENCY?` might return `5.8239462705E+3` indicating that the audio frequency is 5.824 kHz.

FETCh:AUDio:HARMonic:COUNt? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the number of harmonics in the audio spectrum measurement.

The number of harmonics appear in the results table in the audio spectrum view.

Conditions Measurement views: Audio Spectrum

Group Fetch commands

Syntax `FETCh:AUDio:HARMonic:COUNt?`

Related Commands FETCh:AUDio:NHARmonic:COUNt?

Arguments None

Returns <count> ::= <NR1> the number of harmonics in the measurement view.

Examples `FETCH:AUDIO:HARMONIC:COUNT?` might return `3` indicating that there are three harmonics in the measurement.

FETCh:AUDio:HARMonic<x>:AMPLitude? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the amplitude of the specified harmonic in the audio spectrum measurement.

The value <x> represents the specific harmonic, from 1 to the value specified by the CALCulate:AUDio:HARMonic:HNUMber command.

Conditions Measurement views: Audio Spectrum

Group Fetch commands

Syntax `FETCh:AUDio:HARMonic<x>:AMPLitude?`

Related Commands CALCulate:AUDio:HARMonic:HNUMber

Arguments None

Returns <value> ::= <NRf> the amplitude of the specified harmonic in dBc.

Examples `FETCh:AUDio:HARMonic3:AMPLitude?` might return `-2.861` indicating that amplitude of the third harmonic is -2.86 dBc.

FETCh:AUDio:HARMonic<x>:FREQuency? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the frequency of the specified harmonic in the audio spectrum measurement.

The value <x> represents the specific harmonic, from 1 to the value specified by the CALCulate:AUDio:HARMonic:HNUMber command.

Conditions Measurement views: Audio Spectrum

Group Fetch commands

Syntax `FETCh:AUDio:HARMonic<x>:FREQuency?`

Related Commands CALCulate:AUDio:HARMonic:HNUMber

Arguments None

Returns <value> ::= <NRF> the frequency of the specified harmonic.

Examples `FETCh:AUDio:HARMonic3:FREQuency?` might return `17.4718007813E+3` indicating that the frequency of the third harmonic is 17.472 kHz.

FETCh:AUDio:HNOise? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the difference between the current RMS modulation value and the reference value.

This information is only available when the Hum & Noise is measurement is enabled; see the [SENSe]:AUDio:HNOise:ENABle command. The reference value is stored after capturing the reference by pressing the Capture Reference button or by enabling the [SENSe]:AUDio:HNOise:REFerence command.

Conditions Measurement views: Audio Summary

Group Fetch commands

Syntax `FETCh:AUDio:HNOise?`

Related Commands [SENSe]:AUDio:HNOise:ENABle[SENSe]:AUDio:HNOise:REFerence

Arguments None

Returns <value> ::= <NRF> specifies the difference between the Hum & Noise reference and the actual signal in dB.

Examples `FETCH:AUDIO:HNOISE?` might return `0.00 dB` indicating that there is no difference between the Hum & Noise reference and the actual signal.

FETCh:AUDio:HNREFerence? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the Hum and Noise RMS modulation reference value.

The value is stored after capturing the reference by pressing the Capture Reference button or by enabling the [SENSe]:AUDio:HNOise:REFerence command. This information is only available when the Hum & Noise measurement is enabled; see the [SENSe]:AUDio:HNOise:ENABle command.

For AM signal types, the reference represents modulation depth. For FM signal types, the reference represents frequency deviation. For PM signal types the reference represents phase. For Direct signal types the reference represents power.

Conditions Measurement views: Audio Summary

Group Fetch commands

Syntax `FETCh:AUDio:HNREFerence?`

Related Commands [SENSe]:AUDio:HNOise:ENABle[SENSe]:AUDio:HNOise:REFerence

Arguments None

Returns <value> ::= <NRf> the Hum & Noise reference frequency.

Examples `FETCH:AUDIO:HNREFERENCE?` might return `20.581121E+3` indicating that the Hum & Noise reference frequency is 20.58 kHz.

FETCh:AUDio:HPTPeak? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the half peak-to-peak modulation excursion (where the modulation excursion depends on the signal type).

For AM signal types, the modulation excursion is "% Modulation Depth." For FM signal types, the modulation excursion is "Frequency Deviation." For PM signal types, the modulation excursion is "Phase Deviation." For Direct signal types, there is no modulation excursion; it is actually "signal excursion."

Conditions Measurement views: Audio Summary

Group Fetch commands

Syntax `FETCh:AUDio:HPTPeak?`

Arguments None

Returns <value> ::= <NRF> specifies the half peak modulation excursion.

Examples `FETCH:AUDIO:HPTPEAK?` might return `125.28231E+3` indicating that the half peak modulation excursion is 125.28 kHz.

FETCh:AUDio:MODDist? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the modulation distortion for the audio measurement.

Conditions Measurement views: Audio Summary

Group Fetch commands

Syntax `FETCh:AUDio:MODDist?`

Arguments None

Returns <value> ::= <NRf> specifies the modulation distortion.

Examples `FETCH:AUDIO:MODDIST?` might return `98.6282113` indicating that the modulation distortion is 98.628%.

FETCh:AUDio:NHARmonic:COUNt? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the number of non-harmonics in the audio spectrum measurement.

The number of non-harmonics appear in the results table in the audio spectrum view.

Conditions Measurement views: Audio Spectrum

Group Fetch commands

Syntax `FETCh:AUDio:NHARmonic:COUNt?`

Related Commands FETCh:AUDio:HARMonic:COUNt?

Arguments None

Returns <count> ::= <NR1> the number of non-harmonics in the measurement view.

Examples `FETCH:AUDIO:NHARMONIC:COUNT?` might return `7` indicating that there are seven harmonics in the measurement.

FETCh:AUDio:NHARmonic<x>:AMPLitude? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the amplitude of the specified non-harmonic in the audio spectrum measurement.

The value <x> represents the specific non-harmonic, from 1 to the value specified by the CALCulate:AUDio:HARMonic:NHNumber command.

Conditions Measurement views: Audio Spectrum

Group Fetch commands

Syntax `FETCh:AUDio:NHARmonic<x>:AMPLitude?`

Related Commands CALCulate:AUDio:HARMonic:NHNumber

Arguments None

Returns <value> ::= <NRf> the amplitude of the specified non-harmonic in dBc.

Examples `FETCh:AUDio:NHARMonic3:AMPLitude?` might return `-2.861` indicating that amplitude of the third non-harmonic is -2.86 dBc.

FETCh:AUDio:NHARmonic<x>:FREQuency? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the frequency of the specified non-harmonic in the audio spectrum measurement.

The value <x> represents the specific non-harmonic, from 1 to the value specified by the CALCulate:AUDio:HARMonic:NHNumber command.

Conditions Measurement views: Audio Spectrum

Group Fetch commands

Syntax `FETCh:AUDio:NHARmonic<x>:FREQuency?`

Related Commands CALCulate:AUDio:HARMonic:NHNumber

Arguments None

Returns <value> ::= <NRF> the frequency of the specified non-harmonic.

Examples `FETCh:AUDio:NHARMonic3:FREQuency?` might return `17.4718007813E+3` indicating that the frequency of the third non-harmonic is 17.472 kHz.

FETCh:AUDio:PNEGative? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the minus peak modulation excursion (where the modulation excursion depends on the signal type).

For AM signal types, the modulation excursion is "% Modulation Depth." For FM signal types, the modulation excursion is "Frequency Deviation." For PM signal types, the modulation excursion is "Phase Deviation." For Direct signal types, there is no modulation excursion; it is actually "signal excursion."

Conditions Measurement views: Audio Summary

Group Fetch commands

Syntax `FETCh:AUDio:PNEGative?`

Arguments None

Returns <value> ::= <NRf> specifies the minus peak modulation excursion.

Examples `FETCH:AUDIO:PNEGATIVE?` might return `-196.04321E+3` indicating that the minus peak modulation excursion is -196.04 kHz.

FETCh:AUDio:POWer? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the carrier power or signal power depending the signal type.

If the signal type is Direct, the returned value is the average power of the input signal. If the signal type is AM, FM, or PM, the returned value is the carrier power; the average power of the carrier signal with the modulation removed.

Conditions Measurement views: Audio Summary

Group Fetch commands

Syntax `FETCh:AUDio:POWer?`

Arguments None

Returns <value> ::= <NRf> specifies the carrier or signal power in dBm.

Examples `FETCH:AUDIO:POWER?` might return `-22.231123` indicating that the carrier power is -22.23 dBm.

FETCh:AUDio:PPOSitive? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the positive peak modulation excursion (where the modulation excursion depends on the signal type).

For AM signal types, the modulation excursion is “% Modulation Depth.” For FM signal types, the modulation excursion is “Frequency Deviation.” For PM signal types, the modulation excursion is “Phase Deviation.” For Direct signal types, there is no modulation excursion; it is actually “signal excursion.”

Conditions Measurement views: Audio Summary

Group Fetch commands

Syntax `FETCh:AUDio:PPOSitive?`

Arguments None

Returns <value> ::= <NRf> specifies the positive peak modulation excursion.

Examples `FETCH:AUDIO:PPOSITIVE?` might return `215.04321E+3` indicating that the positive peak modulation excursion is -215.04 kHz.

FETCh:AUDio:RMS? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the RMS modulation excursion for the audio measurement (where the modulation excursion depends on the signal type).

For AM signal types, the modulation excursion is "% Modulation Depth." For FM signal types, the modulation excursion is "Frequency Deviation." For PM signal types, the modulation excursion is "Phase Deviation." For Direct signal types, there is no modulation excursion; it is actually "signal excursion."

Conditions Measurement views: Audio Summary

Group Fetch commands

Syntax `FETCh:AUDio:RMS?`

Arguments None

Returns <value> ::= <NRf> specifies the RMS modulation excursion.

Examples `FETCH:AUDIO:RMS?` might return `20.575039E+3` indicating that RMS modulation excursion is 20.575 kHz.

FETCh:AUDio:SINad? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the signal-to-noise and distortion for the audio measurement.

Conditions Measurement views: Audio Summary

Group Fetch commands

Syntax `FETCh:AUDio:SINad?`

Arguments None

Returns <value> ::= <NRf> specifies the RMS modulation excursion.

Examples `FETCH:AUDIO:SINAD?` might return `176.229024E-3` indicating that the signal-to-noise and distortion for the audio measurement is 0.18 dB.

FETCh:AUDio:SNOise? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the signal level to noise level with the harmonic distortion and non-harmonic distortion components removed.

Conditions Measurement views: Audio Summary

Group Fetch commands

Syntax `FETCh:AUDio:SNOise?`

Arguments None

Returns <value> ::= <NRf> specifies the signal level to noise level.

Examples `FETCH:AUDIO:SNOISE?` might return `-12.8156364` indicating the signal level to noise level is -12.82 dB.

FETCh:AUDio:SPECtrum:TRACe<x>? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the audio spectrum trace data for the audio measurement.

The <x> represents trace 1 for this measurement. No other traces are supported.

Conditions Measurement views: Audio Spectrum

Group Fetch commands

Syntax `FETCh:AUDio:SPECtrum:TRACe<x>?`

Arguments None

Returns #<num_digit><num_byte><date(1)><data(2)> . . . <data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the data value at the n^{th} data point in percent (%), 4-byte little endian floating point format specifed in IEEE.488.2.

Examples `FETCh:AUDio:SPECtrum:TRACe1?` might return `#516004xxxx ...` (16004 byte data) for the audio measurement.

FETCh:AUDio:THDist? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the percentage of the total harmonic distortion in the audio measurement.

Conditions Measurement views: Audio Summary

Group Fetch commands

Syntax `FETCh:AUDio:THDist?`

Related Commands FETCh:AUDio:THDist:DB?

Arguments None

Returns <value> ::= <NRf> specifies the percentage of the total harmonic distortion.

Examples `FETCH:AUDIO:THDIST?` might return `53.332921` indicating the percentage of total harmonic distortion is 53.333%.

FETCh:AUDio:THDist:DB? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the total harmonic distortion in dB in the audio measurement.

Conditions Measurement views: Audio Summary

Group Fetch commands

Syntax `FETCh:AUDio:THDist:DB?`

Related Commands FETCh:AUDio:THDist?

Arguments None

Returns <value> ::= <NRf> specifies the total harmonic distortion in dB.

Examples `FETCH:AUDIO:THDIST:DB?` might return `-5.46009` indicating that the total harmonic distortion is -5.46 dB.

FETCh:AUDio:TNHDist? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the percentage of the total non-harmonic distortion in the audio measurement.

Conditions Measurement views: Audio Summary

Group Fetch commands

Syntax `FETCh:AUDio:TNHDist?`

Related Commands FETCh:AUDio:TNHDist:DB?

Arguments None

Returns <value> ::= <NRf> specifies the percentage of the total non-harmonic distortion.

Examples `FETCH:AUDIO:TNHDIST?` might return `297.332921` indicating the percentage of total non-harmonic distortion is 297.333%.

FETCh:AUDio:TNHDist:DB? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the total non-harmonic distortion in dB in the audio measurement.

Conditions Measurement views: Audio Summary

Group Fetch commands

Syntax `FETCh:AUDio:TNHDist:DB?`

Related Commands FETCh:AUDio:TNHDist?

Arguments None

Returns <value> ::= <NRf> specifies the total non-harmonic distortion in dB.

Examples `FETCH:AUDIO:TNHDIST:DB?` might return `9.46009` indicating that the total harmonic distortion is 9.46 dB.

FETCh:AVTime:AVERage? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the RMS (root-mean-square) value for the selected trace in the Amplitude versus Time measurement. Select the trace using the TRACe<x>:AVTime:SELect command.

Conditions Measurement views: Amplitude versus Time

Group Fetch commands

Syntax `FETCh:AVTime:AVERage?`

Arguments None

Returns `<avg> :: <NRf>` is the RMS amplitude in dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `FETCH:AVTIME:AVERAGE?` might return `-2.53`, indicating the RMS amplitude is -2.53 dBm.

FETCh:AVTime:{FIRSt|SECond|THIRd|FOURth}? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the trace data in the Amplitude versus Time measurement.

The mnemonics FIRSt, SECond, THIRd, and FOURth represent Trace 1, Trace 2, Trace 3, and Math trace, respectively. The traces can be specified by the TRACe<x>:AVTime command subgroup.

Conditions Measurement views: Amplitude versus Time

Group Fetch commands

Syntax `FETCh:AVTime:{FIRSt|SECond|THIRd|FOURth}?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the amplitude in dBm at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `FETCH:AVTIME:FIRST?` might return `#3156xxxx...` (156-byte data) for Trace 1.

FETCh:AVTime:MAXimum? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the maximum value for the selected trace in the Amplitude versus Time measurement. Select the trace using the TRACe<x>:AVTime:SELect command.

Conditions Measurement views: Amplitude versus Time

Group Fetch commands

Syntax `FETCh:AVTime:MAXimum?`

Related Commands FETCh:AVTime:MAXLocation?

Arguments None

Returns `<max> :: <NRf>` is the maximum Amplitude in dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `FETCH:AVTIME:MAXIMUM?` might return `-2.84`, indicating the maximum amplitude is -2.84 dBm.

FETCh:AVTime:MAXLocation? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the time at which the amplitude is maximum for the selected trace in the Amplitude versus Time measurement. Select the trace using the TRACe<x>:AVTime:SELect command.

Conditions Measurement views: Amplitude versus Time

Group Fetch commands

Syntax `FETCh:AVTime:MAXLocation?`

Related Commands FETCh:AVTime:MAXimum?

Arguments None

Returns <max_time> :: <NRf> is the time at the maximum in seconds.

Examples FETCH:AVTIME:MAXLOCATION? might return 25.03E-9, indicating the amplitude is maximum at 25.03 ns.

FETCh:AVTime:MINimum? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the minimum value for the selected trace in the Amplitude versus Time measurement. Select the trace using the TRACe<x>:AVTime:SELect command.

Conditions Measurement views: Amplitude versus Time

Group Fetch commands

Syntax FETCh:AVTime:MINimum?

Related Commands FETCh:AVTime:MINLocation?

Arguments None

Returns <min> :: <NRf> is the minimum amplitude in dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples FETCH:AVTIME:MINIMUM? might return -57.64, indicating the minimum amplitude is -57.64 dBm.

FETCh:AVTime:MINLocation? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the time at which the amplitude is minimum for the selected trace in the Amplitude versus Time measurement. Select the trace using the TRACe<x>:AVTime:SELect command.

Conditions Measurement views: Amplitude versus Time

Group Fetch commands

Syntax `FETCh:AVTime:MINLocation?`

Related Commands FETCh:AVTime:MINimum?

Arguments None

Returns `<min_time> ::` `<NRf>` is the time at the minimum in seconds.

Examples `FETCH:AVTIME:MINLOCATION?` might return `450.7E-9`, indicating the amplitude is minimum at 450.7 ns.

FETCh:AVTime:RESult? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the measurement results for the selected trace in the Amplitude versus Time measurement. Select the trace using the TRACe<x>:AVTime:SELect command.

Conditions Measurement views: Amplitude versus Time

Group Fetch commands

Syntax `FETCh:AVTime:RESult?`

Arguments None

Returns `<max>,<max_time>,<min>,<min_time>,<rms>`

Where

<max> :: <NRf> is the maximum amplitude in dBm.
<max_time> :: <NRf> is the time at the maximum in seconds.
<min> :: <NRf> is the minimum amplitude in dBm.
<min_time> :: <NRf> is the time at the minimum in seconds.
<rms> :: <NRf> is the RMS amplitude in dBm.
The unit of amplitude can be changed by the [SENSe]:POWer:UNITs command.

Examples `FETCH:AVTIME:RESULT?` might return `-2.68,48.62E-6,-82.47,22.11E-6,-8.24`, indicating that the maximum amplitude is -2.68 dBm at 48.62 µs, the minimum amplitude is -82.47 dBm at 22.11 µs, and the RMS amplitude is -8.24 dBm.

FETCh:BIBEmissions:FTX? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the band number corresponding to the transmitted FTX detected in the Bluetooth InBand Emission display.

Conditions Measurement views: Bluetooth InBand Emission

Group Fetch commands

Syntax `FETCh:BIBEmissions:FTX?`

Arguments None

Returns <value> ::= <NR1> the band number.

Examples `FETCH:BIBEMISSIONS:FTX?` might returns 39 indicating the band number corresponding to transmitted FTX.

FETCh:BIBEmissions:POWer? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns all channel power results from the Bluetooth InBand Emission display.

Conditions Measurement views: Bluetooth InBand Emission

Group Fetch commands

Syntax `FETCh:BIBEmissions:POWer?`

Arguments None

Returns <Band0_power>,<Band1_power>,<Band2_power>, . . . <Band78_power>

Where:
Bandx_power is the integrated power in that band. All power values are in dBm

Examples `FETCH:BIBEMISSIONS:POWER?` might return 4.227,-28.420,-23.847, . . . ,-29.225, indicating:
(integrated power for Band 0) = 4.227 dBm,
(Integrated power for Band 1) = -28.420 dBm,
(Integrated power for Band 2) = -23.847 dBm, and
(Integrated power for Band 78) = -29.225 dBm.

FETCh:BIBEmissions:RESUlts:STATus? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the Pass/Fail result status of the Bluetooth InBand Emission display.

Conditions Measurement views: Bluetooth InBand Emission

Group Fetch commands

Syntax `FETCh:BIBEmissions:RESUlts:STATus?`

Arguments None

Examples `FETCH:BIBEMISSIONS:RESULTS:STATUS?` returns the Pass/Fail status result on the display.

FETCh:BLUEtooth:CONSte:FERRor? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the frequency error in Hz in the Bluetooth Constellation. The frequency error is the difference between the measured carrier frequency of the signal and the user-selected center frequency.

Conditions Measurement views: Bluetooth Constellation

Group Fetch commands

Syntax `FETCh:BLUEtooth:CONSte:FERRor?`

Arguments None

Returns <freq_error> ::= <NRf> the frequency error.

Examples `FETCh:BLUEtooth:CONSte:FERRor?` Might return -10.7E+3, which is a frequency error of -10.7 kHz.

FETCh:BLUEtooth:CONSte:TRACe? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Queries the Bluetooth Constellation trace data.

Conditions Measurement views: Bluetooth Constellation

Group Fetch commands

Syntax `FETCh:BLUEtooth:CONSte:TRACe?`

Arguments None

Returns #<num_digit><num_byte><I(1)><Q1><I(2)><Q2> . . . <I(n)><Qn>

Where:

<num_digit> is the number of bytes in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<I(n)> and <Qn> are the normalized I- and Q-coordinate values at the nth data point. 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCh:BLUEtooth:CONSte:TRACe?` might return #43848xxxx . . . (3848-byte data) for the trace.

FETCh:BLUEtooth:EDIagram:FDEViation? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the frequency deviation vs. time trace data with the X values in the Bluetooth eye diagram.

Conditions Measurement views: Bluetooth eye diagram.

Group Fetch commands

Syntax `FETCh:BLUEtooth:EDIagram:FDEViation?`

Arguments None

Returns #<num_digits><num_bytes><Y(1)><X(1)><Y(2)><X(2)> . . . <Y(n)><X(n)>

Where

<num_digits> is the number of bytes in <num_bytes>.

<num_bytes> is the number of bytes of data that follow.

<Y(1)> amd <X(1)> are the normalized I- and Q-coordinate values at the n^{th} datapoint. The 4–byte little endian floating point format is specified in IEEE 488.2.

Examples `FETCh:BLUEtooth:EDIagram:FDEViation?` might return #3160xxxx... (160-byte data) for the frequency deviation versus time trace.

FETCh:BLUEtooth:EDIagram:FERRor? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the frequency error in the Bluetooth eye diagram measurement.

Conditions Measurement views: Bluetooth eye diagram.

Group Fetch commands

Syntax `FETCh:BLUEtooth:EDIagram:FERRor?`

Arguments None

Returns <freq_error> ::= <NRf> the frequency error in Hz.

Examples `FETCh:BLUEtooth:EDIagram:FERRor?` might return -10.7E+3, indicating the frequency error is -10.7 kHz.

FETCh:BLUEtooth:EDIagram:HORiz:SCALe? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the value of the horizontal scale in the Bluetooth eye diagram.

Conditions Measurement views: Bluetooth eye diagram.

Group Fetch commands

Syntax `FETCh:BLUEtooth:EDIagram:HORiz:SCALe?`

Arguments None

Returns <value> ::= <NR3> the horizontal scale value.

Examples `FETCH:BLUETOOTH:EDIAGRAM:HORIZ:SCALE?` might return 2 indicating the horizontal scale value is 2 symbols.

FETCh:BLUEtooth:EDIagram:I? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the I vs. time trace in the Bluetooth eye diagram.

Conditions Measurement views: Bluetooth eye diagram.

Group Fetch commands

Syntax `FETCh:BLUEtooth:EDIagram:I?`

Arguments None

Returns #<num_digits><num_bytes><Y(1)><X(1)><Y(2)><X(2)> . . . <Y(n)><X(n)>

Where
<num_digits> is the number of bytes in <num_bytes>.
<num_bytes> is the number of bytes of data that follow.
<Y(1)> amd <X(1)> are the normalized I- and Q-coordinate values at the n^{th} datapoint. The 4–byte little endian floating point format is specified in IEEE 488.2.

Examples `FETCh:BLUEtooth:EDIagram:I?` might return #3160xxxx... (160-byte data) for the I vs. time trace.

FETCh:BLUEtooth:EDIagram:Q? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the Q vs. time trace in the Bluetooth eye diagram.

Conditions Measurement views: Bluetooth eye diagram.

Group Fetch commands

Syntax `FETCh:BLUEtooth:EDIagram:Q?`

Arguments None

Returns #<num_digits><num_bytes><Y(1)><X(1)><Y(2)><X(2)> . . . <Y(n)><X(n)>

Where
<num_digits> is the number of bytes in <num_bytes>.
<num_bytes> is the number of bytes of data that follow.
<Y(1)> amd <X(1)> are the normalized I- and Q-coordinate values at the n^{th} datapoint. The 4–byte little endian floating point format is specified in IEEE 488.2.

Examples `FETCh:BLUEtooth:EDIagram:Q?` might return #3160xxxx... (160-byte data) for the Q vs. time trace.

FETCh:BLUEtooth:FDVTime:TRACe? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the Bluetooth Frequency Deviation vs. Time trace data.

Conditions Measurement views: Frequency Deviation vs. Time

Group Fetch commands

Syntax `FETCh:BLUEtooth:FDVTime:TRACe?`

Arguments None

Returns #<num_digits><num_bytes><data(1)><data(2)> . . . <data(n)>

Where
<num_digits> is the number of bytes in <num_bytes>.
<num_bytes> is the number of bytes of data that follow.
<data(n)> is the frequency deviation in Hz at the n^{th} datapoint. The 4-byte little endian floating point format is specified in IEEE 488.2.

Examples `FETCh:BLUEtooth:FDVTime:TRACe?` might return #3160xxxx... (160-byte data) for the trace.

FETCh:BLUEtooth:FREQuency:ERROr? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the frequency error from the Bluetooth Constellation diagram.

Conditions Measurement views: Bluetooth Constellation

Group Fetch commands

Syntax `FETCh:BLUEtooth:FREQuency:ERROr?`

Arguments None

Returns <NR3>, the frequency error value in Hz.

Examples FETCH:BLUETOOTH:FREQUENCY:ERROR? might return 0.0000 indicating the frequency error is 0.0000 Hz.

FETCh:BLUEtooth:FREQuency:ERROr:TYPE? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the frequency error type in the Bluetooth Constellation display.

Conditions Measurement views: Bluetooth Constellation

Group Fetch commands

Syntax FETCh:BLUEtooth:FREQuency:ERROr:TYPE?

Arguments None

Returns 1 indicates the error is automatically detected.

0 indicates the error is not automatically detected.

Examples FETCH:BLUETOOTH:FREQUENCY:ERROR:TYPE? might return 1 indicating the frequency error is automatically detected.

FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:F1FZero? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the drift in frequency offset (f1–f0 — the frequency offset calculated in the first interval in the payload — the frequency offset calculated in the preamble) in the Bluetooth Frequency Deviation vs. Time display.

Conditions Measurement views: Frequency Deviation vs. Time

Group Fetch commands

Syntax `FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:F1FZero?`

Arguments None

Returns <NR3> the drift in frequency offset (f1–f0 — the frequency offset calculated in the first interval in the payload — the frequency offset calculated in the preamble)

Examples `FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:F1FZero?` might return -207.6465301514 indicating the drift in frequency offset (f1–f0 — the frequency offset calculated in the first interval in the payload — the frequency offset calculated in the preamble)

FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the maximum drift of the frequency offset in payload intervals spaced 50 µs away in the Bluetooth Frequency Deviation vs. Time display.

Conditions Measurement views: Frequency Deviation vs. Time

Group Fetch commands

Syntax `FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5?`

Arguments None

Returns <NR3> the maximum drift of the frequency offset in payload intervals spaced 50 µs away.

Examples `FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNF5?` might return -207.6465301514 indicating the maximum drift of the frequency offset in payload intervals spaced 50 µs away.

FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5:INTErval? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the interval at which the maximum drift of frequency offset in a duration of 50 µs occurred.

Conditions Measurement views: Frequency Deviation vs. Time

Group Fetch commands

Syntax `FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5:INTErval?`

Arguments None

Returns <NR1> interval at which the maximum drift of frequency offset in a duration of 50 µs occurred.

Examples `FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNF5:INTErval?` might return 16, indicating the interval at which the maximum drift of frequency offset in a duration of 50 µs occurred. Here 16 refers to the 16th interval and that the drift of frequency offset calculated in the 16th interval from that calculated in the 11th interval is the maximum among those calculated 50 us away.

FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the maximum drift of the frequency offset of the intervals in the payload from the preamble packet in the Bluetooth Frequency Deviation vs. Time display.

Conditions Measurement views: Frequency Deviation vs. Time

Group Fetch commands

Syntax `FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO?`

Arguments None

Returns <NR3> the maximum drift of frequency offset calculated in payload intervals from the offset calculated in the preamble.

Examples `FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZero?` might return -207.6465301514, indicating that the maximum drift of frequency offset calculated in payload intervals from the offset calculated in the preamble is -207.6465301515 Hz.

FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO:INTErval? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the interval (n) at which the maximum drift fn-f0 occurred in the Bluetooth Frequency Deviation vs. Time display. (The fn-f0 indicates the drift of the frequency offset calculated in each payload interval from the offset calculated in the preamble.)

Conditions Measurement views: Frequency Deviation vs. Time

Group Fetch commands

Syntax `FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO:INTErval?`

Arguments None

Returns <NR1> the interval at which the maximum drift (fn-f0) occurred.

Examples `FETCh:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZero:INTErval?` might return 16, indicating that at the 16th payload interval, the maximum drift occurred. f(16) - f(0) is the maximum drift among all f(n) -f (0).

FETCh:BLUEtooth:FREQuency:OFFSet:MAX? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the maximum frequency offset in the Frequency Deviation vs. Time and the Center Frequency Offset and Drift displays.

Conditions Measurement views: Frequency Deviation vs. Time, CF Offset

Group Fetch commands

Syntax `FETCh:BLUEtooth:FREQuency:OFFSet:MAX?`

Arguments None

Returns <NR3> the maximum frequency offset value.

Examples `FETCh:BLUEtooth:FREQuency:OFFSet:MAX?` might return -189.4632263184, the frequency offset value.

FETCh:BLUEtooth:FREQuency:OFFSet:MAX:INTErval? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the maximum frequency offset interval in the Frequency Deviation vs. Time and the Center Frequency Offset and Drift displays.

Conditions Measurement views: Frequency Deviation vs. Time, CF Offset

Group Fetch commands

Syntax `FETCh:BLUEtooth:FREQuency:OFFSet:MAX:INTErval?`

Arguments None

Returns <NR1> the maximum frequency offset interval.

Examples `FETCh:BLUEtooth:FREQuency:OFFSet:MAX:INTErval?` might return 271 indicating that the maximum drift f(n) - f(0) occurred at the 271st payload interval.

FETCh:BLUEtooth:FREQuency:OFFSet:PREAmble? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the offset calculated in the preamble region in the Frequency Deviation vs. Time and the Center Frequency Offset and Drift displays.

Conditions Measurement views: Frequency Deviation vs. Time, CF Offset

Group Fetch commands

Syntax `FETCh:BLUEtooth:FREQuency:OFFSet:PREAmble?`

Arguments None

Returns <NR3> the offset calculated in the preamble region.

Examples `FETCh:BLUEtooth:FREQuency:OFFSet:PREAmble?` might return 106.4204711914, indicating the offset calculated in the preamble region.

FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:F? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the average frequency value when the selected test pattern is Other in the Bluetooth Frequency Deviation vs. Time display.

Conditions Measurement views: Frequency Deviation vs. Time

Group Fetch commands

Syntax `FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:F?`

Arguments None

Returns <NRf> the average frequency.

Examples `FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:F?` might return 140.8309531250E+3, the average frequency value.

FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:F2F1ratio? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the average ratio value for the high and low frequency deviation in the Bluetooth Frequency Deviation vs. Time display.

Conditions Measurement views: Frequency Deviation vs. Time

Group Fetch commands

Syntax `FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:F2F1ratio?`

Arguments None

Returns <NRf> the average ratio value.

Examples `FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:F2F1ratio?` might return 24.37E-3, the average ratio value.

FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:FONE? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the average frequency deviation for the low deviation pattern in the Bluetooth Frequency Deviation vs. Time display.

Conditions Measurement views: Frequency Deviation vs. Time

Group Fetch commands

Syntax `FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:FONE?`

Arguments None

Examples `FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:FONE?` might return 139.8309531250E+3, the average frequency deviation for the low deviation pattern.

FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:FTWO? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the average frequency deviation for the high deviation pattern in the Bluetooth Frequency Deviation vs. Time display.

Conditions Measurement views: Frequency Deviation vs. Time

Group Fetch commands

Syntax `FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:FTWO?`

Arguments None

Examples `FETCh:BLUEtooth:MODUlation:CHARacteristics:AVErage:FTWO?` might return 139.8309531250E+3, the average frequency deviation for the high deviation pattern.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:COUNt? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the number of packets analyzed to obtain average drift results from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:COUNt?`

Arguments None

Returns <NR1> the number of packets analyzed.

Examples `FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:COUNT?` might return 8, the number of packets analyzed.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the drift f1–f0 from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero?`

Arguments None

Returns <NRf> the drift f1–f0 value.

Examples `FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:F1FZERO?` might return 2.846E+3 indicating that the drift f1–f0 value is 2.846 kHz.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero:STATus? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the Pass/Fail drift f1–f0 result status in the packet from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero:STATus?`

Arguments None

Returns <string> the Pass/Fail result status.

Examples `FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:F1FZERO:STATUS?` might return PASS, FAIL, or NA indicating whether the drift f1–f0 is within the specified limit.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the maximum drift of the fn–fn-5 value from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5?`

Arguments None

Returns <NRf> the fn–fn-5 value.

Examples `FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:FNFN5?` might return 2.846E+3 indicating that the maximum drift fn–fn-5 value in the preamble region is -2.846 kHz.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5:STATus? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the Pass/Fail status of the maximum fn–fn-5 drift in the packet from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5:STATus?`

Arguments None

Returns <string> the Pass/Fail result status.

Examples `FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:FNFN5:STATUS?` might return PASS, FAIL, or NA indicating whether the maximum drift fn–fn-5 is within the specified limit.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the maximum drift fn–f0 from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero?`

Arguments None

Returns <NRf> the fn–fn0 value.

Examples `FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:FNFZERO?` might return 4.846E+3 indicating that the drift fn–f0 value in the preamble region is 4.846 kHz.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero:STATus? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the Pass/Fail result status of the fn–f0 maximum drift in the packet from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero:STATus?`

Arguments None

Returns <string> the Pass/Fail result status.

Examples `FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:FNFZERO:STATUS?` might return PASS, FAIL, or NA indicating whether the maximum drift fn–f0 is within the specified limit.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the maximum frequency offset present in the packet from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX?`

Arguments None

Returns <NRf> the maximum frequency offset value in the packet.

Examples `FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:MAX?` might return 7.446E+3 indicating that the maximum frequency offset value in the packet is 7.446 kHz.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX:STATus? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the Pass/Fail result status of the frequency offset maximum in the packet from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX:STATus?`

Arguments None

Returns <string> the Pass/Fail result status.

Examples `FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:MAX:STATUS?` might return PASS, FAIL, or NA indicating whether the maximum frequency offset in the packet is within the specified limit.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the frequency offset value in the preamble region from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble?`

Arguments None

Returns <NRf> the frequency offset value in the preamble region.

Examples `FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:PREAMBLE?` might return 3.546E+3 indicating that the frequency offset value in the preamble region is 3.546 kHz.

FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble:STATus? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the Pass/Fail result status of the frequency offset in the preamble region.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble:STATus?`

Arguments None

Returns <string> the Pass/Fail result status.

Examples `FETCH:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:PREAMBLE:STATUS?` might return PASS, FAIL, or NA indicating whether the frequency offset in the preamble region is within the specified limit.

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:F:MAX:PERCentage:COUNt? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the number of packets analyzed out of 10 packets for ΔF average from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:F:MAX:PERCentage:COUNt?`

Arguments None

Returns <NR1> the number of packets analyzed.

Examples `FETCH:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:DELTA:F:MAX:PERCENTAGE:COUNT?` might return 3 indicating that three packets were analyzed.

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FONE:MAX:PERCentage:COUNt? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the number of packets analyzed out of 10 packets for the ΔF1avg from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax
```
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FONE:MAX:PERCentage:COUNt?
```

Arguments None

Returns <NR1> the number of packets analyzed.

Examples
```
FETCH:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:DELTA:FONE:MAX:PERCENTAGE:COUNT?
```
might return 6 indicating that six packets were analyzed.

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the percentage of the ΔF2Max values that are above the specified limit.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax
```
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage?
```

Arguments None

Returns <NRf> the number of packets analyzed.

Examples `FETCH:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:DELTA:FTWO:MAX:PERCENTAGE?` might return 99.2 indicating that 99.2% of the deltaF2 maximum values are above the specified limit.

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage:COUNt? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the number of packets analyzed out of 10 packets for the ΔF2Max% from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage:COUNt?`

Arguments None

Returns <NR1> the number of packets analyzed.

Examples `FETCH:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:DELTA:FTWO:MAX:PERCENTAGE:COUNT?` might return 6 indicating that six packets were analyzed.

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage:STATus? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the Pass/Fail result status of the ΔF2Max% field from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage:STATus?`

Arguments None

Returns <string> the Pass/Fail result status.

Examples `FETCH:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:DELTA:FTWO:MAX:PERCENTAGE:STATUS?` might return PASS, FAIL, or NA indicating the Pass/Fail status of the ΔF2Max% field in the Bluetooth summary.

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the ΔFavg value from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F?`

Arguments None

Returns <NRf> the delta F value.

Examples `FETCH:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:F?` might return 2.8723E+3 indicating the delta F average is 2.8723 kHz.

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F2F1ratio? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the ratio of the ΔF2avg to ΔF1avg from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax

```
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:
F2F1ratio?
```

Arguments None

Returns <NRf> the value of the delta F1 to delta F2 ratio.

Examples `FETCH:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:F2F1RATIO?` might return 225.12E+3 indicating the average ratio is 22.5.12 kHz.

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F2F1ratio:STATus? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the Pass/Fail results status of the ΔF2avg/ΔF1avg field from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax

```
FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:
F2F1ratio:STATus?
```

Arguments None

Returns <string> the Pass/Fail result status.

Examples `FETCH:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:F2F1RATIO:STATUS?` might return PASS, FAIL, or NA indicating the Pass/Fall result status of the ΔF2avg/ΔF1avg field.

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FONE? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the ΔF1avg value from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FONE?`

Arguments None

Returns <NRf> the delta F1 value.

Examples `FETCH:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:FONE?` might return 155.3672E+3 indicating the delta F1 average is 155.3672 kHz.

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FONE:STATus? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the Pass/Fail results status of the ΔF1avg from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FONE:STATus?`

Arguments None

Returns <string> the Pass/Fail results status.

Examples `FETCH:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:FONE:STATUS?` might return PASS, FAIL, or NA indicating whether the delta F1 average is within the limit.

FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FTWO? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the ΔF2avg value from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FTWO?`

Arguments None

Returns <NRf> the delta F2 value.

Examples `FETCH:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:FTWO?` might return 225.12E+3 indicating that the delta F2 value is 225.12 kHz.

FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the average output power from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage?`

Arguments None

Returns <NRf> the average output power value in dBm.

Examples `FETCH:BLUETOOTH:SUMMARY:OUTPUT:POWER:AVERAGE?` might return -29.1 indicating that the average output power is -29.1 dBm.

FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage:STATus? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the Pass/Fail results status of the average output power from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage:STATus?`

Arguments None

Returns <string> the Pass/Fail results status.

Examples `FETCH:BLUETOOTH:SUMMARY:OUTPUT:POWER:AVERAGE:STATUS?` might return PASS, FAIL, or NA indicating whether the average output power is within the specified limit.

FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:COUNt? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the number of packets analyzed out of 10 packets for the calculation of the output power from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:COUNt?`

Arguments None

Returns <NR1> the number of packets analyzed.

Examples `FETCH:BLUETOOTH:SUMMARY:OUTPUT:POWER:COUNT?` might return 6 indicating that six packets were analyzed.

FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:COUNt? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the number of packets analyzed out of 10 packets for calculation of the output power for EDR from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:COUNt?`

Arguments None

Returns <NR1> the number of packets analyzed.

Examples `FETCH:BLUETOOTH:SUMMARY:OUTPUT:POWER:EDR:COUNT?` might return 6 indicating that six packets were analyzed.

FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:DPSK? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the power in the DPSK portion of the EDR burst from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:DPSK?`

Arguments None

Returns <NRf> the power in the DPSK portion of the EDR burst.

Examples `FETCH:BLUETOOTH:SUMMARY:OUTPUT:POWER:EDR:DPSK?` might return -29.2 indicating that the average output power is -29.2 dBm.

FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:GPSK? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the power in the GFSK portion of the EDR burst from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:GPSK?`

Arguments None

Returns <NRf> the power in the GFSK portion of the EDR burst.

Examples `FETCH:BLUETOOTH:SUMMARY:OUTPUT:POWER:EDR:GPSK?` might return -29.1 indicating that the average output power is -29.1 dBm.

FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the peak output power from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK?`

Arguments None

Returns <NRf> the peak output power value in dBm.

Examples `FETCH:BLUETOOTH:SUMMARY:OUTPUT:POWER:PEAK?` might return -29.3 indicating that the peak output power is -29.3 dBm.

FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK:STATus? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the Pass/Fail results status of the peak output power from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK:STATus?`

Arguments None

Returns <string> the Pass/Fail result status.

Examples FETCH:BLUETOOTH:SUMMARY:OUTPUT:POWER:PEAK:STATUS? might return PASS, FAIL, or NA indicating whether the peak output power is within the specified limit.

FETCh:BLUEtooth:SUMMary:PACKet:BDR:CRC? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the Basic Rate value of the CRC field from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax FETCh:BLUEtooth:SUMMary:PACKet:BDR:CRC?

Arguments None

Returns <string> the value of the CRC field.

Examples FETCh:BLUEtooth:SUMMary:PACKet:Basic Rate:CRC? Might return "0x9DB0" indicating the value of the CRC field in hexadecimal.

FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:ARQN? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the value of ARQN field from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:ARQN?

Arguments None

Returns <NR2> the value of the ARQN field.

Examples `FETCh:BLUEtooth:SUMMary:PACKet:Basic Rate:HEADer:ARQN?` might return 0.000000 indicating the value of the flow field.

FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:FLOW? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the value of Flow field from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:FLOW?`

Arguments None

Returns <NR2> the flow value.

Examples `FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:FLOW?` might return 1.000000 indicating the value of the flow field.

FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:HEC? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the value of HEC field from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:HEC?`

Arguments None

Returns <string> the value of the HEC field.

Examples `FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:HEC?` might return "00110010" indicating the value of the HEC field in binary.

FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:LT:ADDRess? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the LT address from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:LT:ADDRess?`

Arguments None

Returns <string> the LT address.

Examples `FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:LT:ADDR?` might return "101" indicating the LT address in binary.

FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:SEQN? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the value of SEQN field from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:SEQN?`

Arguments None

Returns <NR2> the value of the SEQN field.

Examples `FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:SEQN?` might return 1.000000 indicating the value of the flow field.

FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:TYPE? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the value of Type field from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:TYPE?`

Arguments None

Returns <string> the value of the type field.

Examples `FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:TYPE?` might return "00001111" indicating the type field in binary.

FETCh:BLUEtooth:SUMMary:PACKet:BDR:PAYLoad:LENgth? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the Basic Rate payload length from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:PACKet:BDR:PAYLoad:LENgth?`

Arguments None

Returns <string> the value of the payload length.

Examples `FETCh:BLUEtooth:SUMMary:PACKet:BDR:PAYLoad:LENgth?` might return "0101010011" indicating the payload length value in binary.

FETCh:BLUEtooth:SUMMary:PACKet:BDR:PREAmble? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the preamble of the Basic Rate packet from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:PACKet:BDR:PREAmble?`

Arguments None

Returns <string> the preamble of the packet.

Examples `FETCh:BLUEtooth:SUMMary:PACKet:BDR:PREAmble?` might return "0101" indicating the preamble is 0101 in binary.

FETCh:BLUEtooth:SUMMary:PACKet:BDR:SYNC:WORD? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the sync word from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:PACKet:BDR:SYNC:WORD?`

Arguments None

Returns <string> the sync word.

Examples FETCh:BLUEtooth:SUMMary:PACKet:BDR:SYNC:WORD? might return "Ox4F36F2CEE85390CB" indicating the sync word in hexadecimal format.

FETCh:BLUEtooth:SUMMary:PACKet:BDR:TYPE? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the packet type of the Basic Rate signal from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax FETCh:BLUEtooth:SUMMary:PACKet:BDR:TYPE?

Arguments None

Returns <string> the packet type.

Examples FETCh:BLUEtooth:SUMMary:PACKet:BDR:TYPE? might return "DH5" indicating that the DH5 packet is being analyzed.

FETCh:BLUEtooth:SUMMary:PACKet:BLE:ACCEss:ADDRess? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the 32-bit Access Address from the Bluetooth Summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax FETCh:BLUEtooth:SUMMary:PACKet:BLE:ACCEss:ADDRess?

Arguments None

Returns <string> the 32-bit Access Address.

Examples FETCH:BLUETOOTH:SUMMARY:PACKET:BLE:ACCESS:ADDRESS? might return "0x71764129" indicating the Access Address in hexadecimal.

FETCh:BLUEtooth:SUMMary:PACKet:BLE:CRC? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the value of the CRC field from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax FETCh:BLUEtooth:SUMMary:PACKet:BLE:CRC?

Arguments None

Returns <string> the value of the CRC field.

Examples FETCH:BLUETOOTH:SUMMARY:PACKET:BLE:CRC? might return "0xAAAAAA" indicating value CRC field in hexadecimal.

FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:LENGth? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the value of the length field from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:LENGth?

Arguments None

Returns <string> the value of the length field.

Examples FETCH:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:LENGTH? might return “100101” indicating value length field in binary.

FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:RX:ADDRess? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the Rx address from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:RX:ADDRess?

Arguments None

Returns <string> the Rx address.

Examples FETCH:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:RX:ADDRESS? might return 0.00000000 indicating Rx Adress.

FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TX:ADDRess? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the Tx address from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TX:ADDRess?

Arguments None

Returns <string> the Tx address.

Examples FETCH:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:TX:ADDRESS? might return 0.00000000 indicating Tx Adress.

FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TYPE? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the PDU type from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax FETCh:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TYPE?

Arguments None

Returns <string> the PDU type.

Examples FETCH:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:TYPE? might return "0010" indicating the PDU type in binary.

FETCh:BLUEtooth:SUMMary:PACKet:BLE:PREAmble? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the preamble of the BLE packet from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax FETCh:BLUEtooth:SUMMary:PACKet:BLE:PREAmble?

Arguments None

Returns <string> the preamble of the packet.

Examples `FETCH:BLUETOOTH:SUMMARY:PACKET:BLE:PREAMBLE?` might return "01010101" indicating the preamble is 01010101 in binary.

FETCh:BLUEtooth:SUMMary:PACKet:BLE:TYPE? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the packet type from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Fetch commands

Syntax `FETCh:BLUEtooth:SUMMary:PACKet:BLE:TYPE?`

Arguments None

Returns <string> the BLE packet type.

Examples `FETCH:BLUETOOTH:SUMMARY:PACKET:BLE:TYPE?` might return "BLE_TEST" indicating the BLE test signal is being analyzed.

FETCh:CCDF? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the CCDF measurement results.

Conditions Measurement views: CCDF

Group Fetch commands

Syntax `FETCh:CCDF?`

Related Commands

Arguments

None

Returns

<avg_ampl>,<avg_ccdf>,<crest_factor>,<ampl_10>,<ampl_1>,
<ampl_p1>,<ampl_p01>,<ampl_p001>,<ampl_p0001>

Where
<avg_ampl> is the average amplitude in dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.
<avg_ccdf> is the average CCDF in percent.
<crest_factor> is the crest factor in dB.
<ampl_10> is the amplitude at CCDF of 10% in dB.
<ampl_1> is the amplitude at CCDF of 1% in dB.
<ampl_p1> is the amplitude at CCDF of 0.1% in dB.
<ampl_p01> is the amplitude at CCDF of 0.01% in dB.
<ampl_p001> is the amplitude at CCDF of 0.001% in dB.
<ampl_p0001> is the amplitude at CCDF of 0.0001% in dB.

Examples

FETCH:CCDF? might return
-33.35,35.8,9.75,3.88,7.07,8.50,9.25,9.72,9.74, indicating
(average amplitude) = -33.35 dBm,
(average CCDF) = 35.8%,
(crest factor) = 9.75 dB,
(amplitude at CCDF of 10%) = 3.88 dB,
(amplitude at CCDF of 1%) = 7.07 dB,
(amplitude at CCDF of 0.1%) = 8.50 dB,
(amplitude at CCDF of 0.01%) = 9.25 dB,
(amplitude at CCDF of 0.001%) = 9.72 dB, and
(amplitude at CCDF of 0.0001%) = 9.74 dB.

FETCh:CCDF:{FIRSt|SECond|THIRd}:X? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the horizontal values of the specified trace in the CCDF measurement.

The mnemonics FIRSt, SECond, and THIRd represent Trace 1, Trace 2, and Gaussian reference curve, respectively.

Conditions

Measurement views: CCDF

Group

Fetch commands

Syntax `FETCh:CCDF:{FIRSt|SECond|THIRd}:X?`

Arguments None

Returns `#<num_digit><num_byte><x(1)><x(2)>...<x(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<x(n)>` is the horizontal value (dB) of the CCDF graph at the n^{th} point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:CCDF:FIRST:X?` might return `#41024xxxx...` (1024-byte data) for the horizontal values of Trace 1.

FETCh:CCDF:{FIRSt|SECond|THIRd}:XY? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the horizontal and vertical value pairs of the specified trace in the CCDF measurement.

The mnemonics FIRSt, SECond, and THIRd represent Trace 1, Trace 2, and Gaussian reference curve, respectively.

Conditions Measurement views: CCDF

Group Fetch commands

Syntax `FETCh:CCDF:{FIRSt|SECond|THIRd}:XY?`

Arguments None

Returns `#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...`
`<x(n)><y(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<x(n)><y(n)>` is the horizontal value (dB) and vertical value (%) pair at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:CCDF:FIRST:XY?` might return `#41024xxxx...` (1024-byte data) for the horizontal and vertical value pairs of Trace 1.

FETCh:CCDF:{FIRSt|SECond|THIRd}[:Y]? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the vertical values of the specified trace in the CCDF measurement.

The mnemonics FIRSt, SECond, and THIRd represent Trace 1, Trace 2, and Gaussian reference curve, respectively.

Conditions Measurement views: CCDF

Group Fetch commands

Syntax `FETCh:CCDF:{FIRSt|SECond|THIRd}[:Y]?`

Arguments None

Returns `#<num_digit><num_byte><y(1)><y(2)>...<y(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<y(n)>` is the vertical value (%) of the CCDF graph at the n^{th} point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:CCDF:FIRST:Y?` might return `#41024xxxx...` (1024-byte data) for the vertical values of Trace 1.

FETCh:CONSte:FERRor? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the frequency error in Hz. The frequency error is the difference between the measured carrier frequency of the signal and the user-selected center frequency of the analyzer.

Group Fetch commands

Syntax `FETCh:CONSte:FERRor?`

Related Commands FETCh:EVM:FERRor?

Arguments None.

Returns `<freq_error> ::` `<NRf>` which is the frequency error in Hz.

Examples `FETCh:CONSte:?` might return -10.7E+3, which is a frequency error of -10.7 kHz.

FETCh:CONSte:RESults? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the constellation measurement results of EVM RMS, peak and location displayed on the bottom of the screen.

Conditions Measurement views: Constellation

Group Fetch commands

Syntax `FETCh:CONSte:RESults?`

Arguments None

Returns For modulation types 2|4|8|16FSK or C4FM:

`FSK_deviation_Avg_Leftmost, FSK_deviation_Avg_Rightmost`

Where
`FSK_deviation_Avg_Leftmost` is the average FSK deviation of the left-most symbol in Hz.
`FSK_deviation_Avg_Rightmost` is the average FSK deviation of the right-most symbol in Hz.

For all other valid modulation types:

`<EVM_RMS>,<EVM_peak>,<location>`

Where
`<EVM_RMS> ::` `<NRf>` is the RMS EVM in percent (%).

<EVM_peak> :: <NRf> is the peak EVM in percent (%).
<location> :: <NRf> is the peak location in symbol number.
The time unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples FETCH:CONSTE:RESULTS? might return 2.841,3.227,68.000, indicating that the RMS EVM is 2.841% and the peak EVM is 3.227% at symbol #68.

FETCh:CONSte:TRACe? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the constellation trace data.

Conditions Measurement views: Constellation

Group Fetch commands

Syntax FETCh:CONSte:TRACe?

Arguments None

Returns #<num_digit><num_byte><I(1)><Q(1)><I(2)><Q(2)>...
<I(n)><Q(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<I(n)> and <Q(n)> are the normalized I- and Q-coordinate values at the n^{th} data point. 4-byte little endian floating-point format specified in IEEE 488.2.

Examples FETCH:CONSTE:TRACE? might return #43848xxxx... (3848-byte data) for the constellation trace data.

FETCh:DDEMod:STABle? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the symbol table data.

Conditions Measurement views: Symbol table

Group Fetch commands

Syntax `FETCh:DDEMod:STABle?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the symbol table data at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:DDEMOD:STABLE?` might return `#3512xxxx...` (512-byte data) for the symbol table.

FETCh:DDEMod:SYNCh:WORD:LENGth? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the length of the synch word in the symbol table.

Conditions Measurement views: Symbol table

Group Fetch commands

Syntax `FETCh:DDEMod:SYNCh:WORD:LENGth?`

Related Commands FETCh:DDEMod:SYNCh:WORD:POSition?

Arguments None

Returns `<NR1>` indicates the length of the synch word in symbols.

Examples `FETCH:DDEMOD:SYNCH:WORD:LENGTH?` might return `3`, indicating the length of the synch word is three symbols.

FETCh:DDEMod:SYNCh:WORD:POSition? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the position of the synch word in the symbol table.

Conditions Measurement views: Symbol table

Group Fetch commands

Syntax `FETCh:DDEMod:SYNCh:WORD:POSition?`

Related Commands FETCh:DDEMod:SYNCh:WORD:LENGth?

Arguments None

Returns `<NR1>` indicates what symbol number the synch word begins at in the table. Zero (0) represents the first symbol in the table. A "-1" indicates that the synch word was not found.

Examples `FETCH:DDEMOD:SYNCH:WORD:POSITION?` might return `10`, indicating the synch word begins at 11th symbol in the table.

FETCh:DIQVtime:FERRor? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the frequency error in the Demod I&Q versus Time measurement.

Conditions Measurement views: Demod I&Q versus Time

Group Fetch commands

Syntax `FETCh:DIQVtime:FERRor?`

Arguments None

Returns `<freq_error> ::` `<NRf>` is the frequency error in Hz.

Examples `FETCH:DIQVTIME:FERROR?` might return `-10.7E+3`, indicating the frequency error is -10.7 kHz.

FETCh:DIQVtime:I? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the I versus Time trace data.

Conditions Measurement views: Demod I&Q versus Time

Group Fetch commands

Syntax `FETCh:DIQVtime:I?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the I level in volts at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:DIQVTIME:I?` might return `#3160xxxx...` (160-byte data) for the I versus Time trace.

FETCh:DIQVtime:Q? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the Q versus Time trace data.

Conditions Measurement views: Demod I&Q versus Time

Group Fetch commands

Syntax `FETCh:DIQVtime:Q?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the Q level in volts at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:DIQVTIME:Q?` might return `#3160xxxx...` (160-byte data) for the Q versus Time trace.

FETCh:DPX:DDENsity? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the calculated density for the selected region in the DPXogram measurement.

Conditions Measurement views: DPX spectrum

Group Fetch commands

Syntax `FETCh:DPX:DDENsity?`

Arguments None

Returns <NR1> value.

Examples `FETCH:DPX:DDENSITY?` might return a DPX signal density value of 24.4802413393.

FETCh:DPX:DGRam:LINE:COUNt? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the number of lines in the DPXogram measurement.

Conditions Measurement views: DPX spectrum

Group Fetch commands

Syntax `FETCh:DPX:DGRam:LINE:COUNt?`

Arguments None

Returns <NR1> number of lines in the measurement.

Examples `FETCH:DPX:DGRAM:LINE:COUNT?` might return `40` indicating that there were 40 lines in the measurement.

FETCh:DPX:DGRam:TIME[:SCALe]:OFFSet? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns the time scale offset in the DPXogram measurement.

Conditions Measurement views: DPX spectrum

Group Fetch commands

Syntax `FETCh:DPX:DGRam:TIME[:SCALe]:OFFSet?`

Arguments None

Returns <string> The time offset in the DPXogram measurement in seconds.

Examples `FETCH:DPX:DGRAM:TIME[:SCALE]:OFFSET?` might return "`0.000 s`" indicating that the time scale offset was 0.000 s.

FETCh:DPX:RESults:TRACe<x>? (Query Only)

This command does not apply to SignalVu-PC or connected instruments.

Returns waveform data of specified trace <x> in the DPX spectrum measurement, where x is 1 to 5. The traces 1–4 are in the standard form. Trace 5 is the bitmap trace and its data is returned in a binary block. Trace 6 is the DPXogram trace on DPX spectrum plots. Trace 7 is the Ogram line in the DPXogram display.

Conditions Measurement views: DPX spectrum

Group Fetch commands

Syntax `FETCh:DPX:RESults:TRACe<x>?`

Arguments <NR1>

Returns For traces 1 to 4:
`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the amplitude of the trace sample at the n^{th} data point.
<data(n) is in the 4-byte little endian floating-point format specified in IEEE 488.2.

The format is a binary block of 32 bit floating point values. Each value ranges from 0 to 1.0, where 1.0 represents 100%.

Examples `FETCH:DPX:RESULTS:TRACE1?` might return `#42004xxxx...` (2004-byte of data) for the waveform data of trace one (1).

FETCh:DPX:TDM:FREQuency:TXBer:BER? (Query Only)

Returns the Tx BER measurement BER results.

Conditions Measurement views: DPX:Frequency

Group Fetch commands

Syntax `FETCh:DPX:TDM:FREQuency:TXBer:BER?`

Related Commands

Arguments None

Returns <NR1> is the calculated BER.

Examples `FETCH:DPX:TDM:FREQUENCY:TXBER:BER?` might return 9.1135684130E-6.

FETCh:DPX:TDM:FREQuency:TXBer:BITS? (Query Only)

Returns the number of bits counted in the active Tx BER measurement session.

Conditions Measurement views: DPX:Frequency

Group Fetch commands

Syntax `FETCh:DPX:TDM:FREQuency:TXBer:BITS?`

Related Commands

Arguments None

Returns <NR1> is the number of bits counted in the current Tx BER measurement session.

Examples `FETCH:DPX:TDM:FREQUENCY:TXBER:BITS?` might return 72.0820000000E+12.

FETCh:DPX:TDM:FREQuency:TXBer:ERRors? (Query Only)

Returns the number of errors counted in the active Tx BER measurement session.

Conditions Measurement views: DPX:Frequency

Group Fetch commands

Syntax `FETCh:DPX:TDM:FREQuency:TXBer:ERRors?`

Arguments None

Returns <NR1> is the number of errors counted in the current Tx BER measurement session.

Examples FETCH:DPX:TDM:FREQUENCY:TXBER:ERRORS? might return 1.0000000000.

FETCh:DPX:TDM:FREQuency:TXBer:STATe? (Query Only)

Returns the state of the Tx BER measurement.

Conditions Measurement views: DPX:Frequency

Group Fetch commands

Syntax FETCh:DPX:TDM:FREQuency:TXBer:STATe?

Returns IDLE indicates the Tx BER measurement is not running.
ACTIVE indicates the Tx BER measurement is in progress.
SYNC indicates the instrument is in the process of syncing with the data pattern.
DONE indicates the Tx BER measurement is finished.

Examples FETCH:DPX:TDM:FREQUENCY:TXBER:STATE? might return ACTIVE indicating the instrument is measuring the Tx BER.

FETCh:DPX:TRACe:AVERage? (Query Only)

Returns waveform data of the average trace in the DPX spectrum measurement.

Conditions Measurement views: DPX spectrum

Group Fetch commands

Syntax FETCh:DPX:TRACe:AVERage?

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the amplitude of the average trace at the n^{th} data point in dBm,

4-byte little endian floating-point format specified in IEEE 488.2. The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples FETCH:DPX:TRACE:AVERAGE? might return #42004xxxx... (2004-byte data) for the waveform data of the average trace.

FETCh:DPX:TRACe:BITMap? (Query Only)

Returns a data block of the DPX spectrum measurement bitmap waveform data.

Conditions Measurement views: DPX spectrum

Group Fetch commands

Syntax FETCh:DPX:TRACe:BITMap?

Arguments None

Returns A binary block of 32-bit floating point values (IEEE standard single precision floating point format). Each value ranges from 0 to 1.0, where 1.0 represents 100%. Bitmaps have 201 rows and 801 columns. Data[0] is the first column in the first row, data[801] is the first column in the 2nd row, and so on.

Bitmap values are returned in the following order: Top left pixel first, then the rest of the top row, after that row by row. In other words, the pixels in the bitmap are returned from left to right and top to bottom; one row at a time.

Examples FETCH:DPX:TRACE:BITMAP? might return #6644004xxxx...

FETCh:DPX:TRACe:MATH? (Query Only)

Returns waveform data of the math trace in the DPX spectrum measurement.

Conditions Measurement views: DPX spectrum

Group Fetch commands

Syntax FETCh:DPX:TRACe:MATH?

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the amplitude of the math trace at the n^{th} data point in dBm, 4-byte little endian floating-point format specified in IEEE 488.2.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `FETCH:DPX:TRACE:MATH?` might return `#42004xxxx...` (2004-byte data) for the waveform data of the math trace.

FETCh:DPX:TRACe:MAXimum? (Query Only)

Returns waveform data of the maximum trace in the DPX spectrum measurement.

Conditions Measurement views: DPX spectrum

Group Fetch commands

Syntax `FETCh:DPX:TRACe:MAXimum?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the amplitude of the maximum trace at the n^{th} data point in dBm, 4-byte little endian floating-point format specified in IEEE 488.2.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `FETCH:DPX:TRACE:MAXIMUM?` might return `#42004xxxx...` (2004-byte data) for the waveform data of the maximum trace.

FETCh:DPX:TRACe:MINimum? (Query Only)

Returns waveform data of the minimum trace in the DPX spectrum measurement.

Conditions Measurement views: DPX spectrum

Group Fetch commands

Syntax `FETCh:DPX:TRACe:MINimum?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the amplitude data of the minimum trace at the n^{th} data point in dBm,
4-byte little endian floating-point format specified in IEEE 488.2.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `FETCH:DPX:TRACE:MINIMUM?` might return `#42004xxxx...` (2004-byte data) for the waveform data of the minimum trace.

FETCh:EDIagram:FDEViation? (Query Only)

Returns the frequency deviation versus Time trace data with the X values.

Group Fetch commands

Syntax `FETCh:EDIagram:FDEViation?`

Related Commands

Returns `#<num_digit><num_byte><Y(1)><X(1)><Y(2)><X(2)>...<Y(n)><X(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.

<Y(n)> and <X(n)> is the frequency deviation in Hz and time (symbols) pair at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples FETCH:EDIAGRAM:FDEVIATION? might return #3160xxxx... (160-byte data) for the frequency deviation versus Time trace.

FETCh:EDIagram:FERRor? (Query Only)

Returns the frequency error in the eye diagram measurement.

Conditions Measurement views: Eye diagram

Group Fetch commands

Syntax FETCh:EDIagram:FERRor?

Arguments None

Returns <freq_error> :: <NRf> is the frequency error in Hz.

Examples FETCH:EDIAGRAM:FERROR? might return -10.7E+3, indicating the frequency error is -10.7 kHz.

FETCh:EDIagram:I? (Query Only)

Returns the I versus Time trace data in the eye diagram measurement.

Conditions Measurement views: Eye diagram

Group Fetch commands

Syntax FETCh:EDIagram:I?

Arguments None

Returns `#<num_digit><num_byte><Y(1)><X(1)><Y(2)><X(2)>...<Y(n)><X(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<Y(n)><X(n)>` is the I level (normalized) and time (symbols) pair at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:EDIAGRAM:I?` might return `#3160xxxx...` (160-byte data) for the I versus Time trace.

FETCh:EDIagram:Q? (Query Only)

Returns the Q versus Time trace data in the eye diagram measurement.

Conditions Measurement views: Eye diagram

Group Fetch commands

Syntax `FETCh:EDIagram:Q?`

Arguments None

Returns `#<num_digit><num_byte><Y(1)><X(1)><Y(2)><X(2)>...<Y(n)><X(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<Y(n)><X(n)>` is the Q level (normalized) and time (symbols) pair at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:EDIAGRAM:Q?` might return `#3160xxxx...` (160-byte data) for the Q versus Time trace.

FETCh:EVM:FERRor? (Query Only)

Returns the frequency error in the EVM versus Time measurement.

Conditions Measurement views: EVM versus Time

Group Fetch commands

Syntax `FETCh:EVM:FERRor?`

Arguments None

Returns `<freq_error> ::` `<NRf>` is the frequency error in Hz.

Examples `FETCH:EVM:FERROR?` might return `-10.7E+3`, indicating the frequency error is -10.7 kHz.

FETCh:EVM:PEAK? (Query Only)

Returns the peak value in the EVM versus Time measurement.

Conditions Measurement views: EVM versus Time

Group Fetch commands

Syntax `FETCh:EVM:PEAK?`

Related Commands FETCh:EVM:PINDex?

Arguments None

Returns `<peak> ::` `<NRf>` is the peak EVM value in percent (%).

Examples `FETCH:EVM:PEAK?` might return `1.32`, indicating the peak EVM value is 1.32%.

FETCh:EVM:PINDex? (Query Only)

Returns the time at the EVM peak.

Conditions Measurement views: EVM versus Time

Group Fetch commands

Syntax `FETCh:EVM:PINDex?`

Related Commands FETCh:EVM:PEAK?

Arguments None

Returns `<peak_time> ::` `<NRf>` is the time at the EVM peak in symbol number. The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples `FETCH:EVM:PINDEX?` might return `68.000`, indicating that the EVM peak is at symbol #68.

FETCh:EVM:RMS? (Query Only)

Returns the RMS (Root-Mean-Square) value in the EVM versus Time measurement.

Conditions Measurement views: EVM versus Time

Group Fetch commands

Syntax `FETCh:EVM:RMS?`

Arguments None

Returns `<rms> ::` `<NRf>` is the RMS EVM value in percent (%).

Examples `FETCH:EVM:RMS?` might return `0.582`, indicating the RMS EVM value is 0.582%.

FETCh:EVM:TRACe? (Query Only)

Returns the EVM versus Time trace data.

Conditions Measurement views: EVM versus Time

Group Fetch commands

Syntax `FETCh:EVM:TRACe?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the EVM value at the n^{th} data point in percent (%),
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:EVM:TRACE?` might return `#42036xxxx...` (2036-byte data) for the EVM versus Time trace.

FETCh:FDVTime:FERRor? (Query Only)

Returns the frequency error in the Frequency deviation versus Time measurement.

Conditions Measurement views: Frequency deviation versus Time

Group Fetch commands

Syntax `FETCh:FDVTime:FERRor?`

Arguments None

Returns <freq_error> :: <NRf> is the frequency error in Hz.

Examples FETCH:FDVTIME:FERROR? might return -10.7E+3, indicating the frequency error is -10.7 kHz.

FETCh:FDVTime:TRACe? (Query Only)

Returns the Frequency deviation versus Time trace data.

Conditions Measurement views: Frequency deviation versus Time

Group Fetch commands

Syntax FETCh:FDVTime:TRACe?

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the frequency deviation in Hz at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples FETCH:FDVTIME:TRACE? might return #3160xxxx... (160-byte data) for the Frequency deviation versus Time trace.

FETCh:{FM|PM}:FERRor? (Query Only)

Returns the frequency error in the Frequency modulation or Phase modulation measurements.

Conditions Measurement views: Frequency deviation versus Time

Group Fetch commands

Syntax FETCh:{FM|PM}:FERRor?

Arguments None

Returns `<freq_error> ::` `<NRf>` is the frequency error in Hz.

Examples `FETCH:{FM|PM}:FERROR?` might return `-10.7E+3`, indicating the frequency error is -10.7 kHz.

FETCh:FM:PHALf? (Query Only)

Returns the half peak-peak frequency deviation (Pk-Pk/2) in the FM measurement.

Conditions Measurement views: FM

Group Fetch commands

Syntax `FETCh:FM:PHALf?`

Arguments None

Returns `<Pk-Pk/2> ::` `<NRf>` is the half peak-peak frequency deviation in Hz.

Examples `FETCH:FM:PHALF?` might return `628.9E+3`, indicating the half peak-peak frequency deviation is 628.9 kHz.

FETCh:FM:PNEGative? (Query Only)

Returns the negative peak frequency deviation (–Pk) in the FM measurement.

Conditions Measurement views: FM

Group Fetch commands

Syntax `FETCh:FM:PNEGative?`

Arguments None

Returns <-Pk> :: <NRf> is the negative peak frequency deviation in Hz.

Examples FETCH:FM:PNEGATIVE? might return –495.6E+3, indicating the negative peak frequency deviation is –495.6 kHz.

FETCh:FM:PPOSitive? (Query Only)

Returns the positive peak frequency deviation (+Pk) in the FM measurement.

Conditions Measurement views: FM

Group Fetch commands

Syntax FETCh:FM:PPOSitive?

Arguments None

Returns <+Pk> :: <NRf> is the positive peak frequency deviation in Hz.

Examples FETCH:FM:PPOSITIVE? might return 763.2E+3, indicating the positive peak frequency deviation is 763.2 kHz.

FETCh:FM:PTPeak? (Query Only)

Returns the peak-peak frequency deviation (Pk-Pk) in the FM measurement.

Conditions Measurement views: FM

Group Fetch commands

Syntax FETCh:FM:PTPeak?

Arguments None

Returns <Pk-Pk> :: <NRf> is the peak-peak frequency deviation in Hz.

Examples FETCH:FM:PTPEAK? might return 1.258E+6, indicating the peak-peak frequency deviation is 1.258 MHz.

FETCh:FM:RESult? (Query Only)

Returns the FM measurement results.

Conditions Measurement views: FM

Group Fetch commands

Syntax FETCh:FM:RESult?

Arguments None

Returns <+Pk>,<-Pk>,<RMS>,<Pk-Pk>,<Pk-Pk/2>

Where
<+Pk> :: <NRf> is the positive peak frequency deviation in Hz.
<-Pk> :: <NRf> is the negative peak frequency deviation in Hz.
<RMS> :: <NRf> is the RMS frequency deviation in Hz.
<Pk-Pk> :: <NRf> is the peak-peak frequency deviation in Hz.
<Pk-Pk/2> :: <NRf> is the half peak-peak frequency deviation in Hz.

Examples FETCH:FM:RESULT? might return
763.2E+3,-494.6E+3,271.2E+3,1.258E+6,628.9E+3.

FETCh:FM:RMS? (Query Only)

Returns the RMS frequency deviation in the FM measurement.

Conditions Measurement views: FM

Group Fetch commands

Syntax FETCh:FM:RMS?

Arguments None

Returns <RMS> :: <NRf> is the RMS frequency deviation in Hz.

Examples FETCH:FM:RMS? might return 271.2E+3, indicating the RMS frequency deviation is 271.2 kHz.

FETCh:{FSETtling|PSETtling}:FTTime? (Query Only)

Returns the settling time from the trigger position in seconds. The settling time is measured from the trigger point (see :FETCh:FSETtling:TRIGger:TIME).

Conditions Measurement views: Frequency and Phase Settling

Group Fetch commands

Syntax FETCh:{FSETtling|PSETtling}:FTTime?

Arguments None

Returns <value> :: <NRf> is the settling time in seconds.

Examples FETCH:FSETTLING:FTTIME? might return 44.8300E-6, indicating the settling time is 44.83 µs.

FETCh:{FSETtling|PSETtling}:MASK[:PASS]? (Query Only)

Returns whether the input signal passes the mask test.

Conditions Measurement views: Frequency and Phase Settling

Group Fetch commands

Syntax FETCh:{FSETtling|PSETtling}:MASK[:PASS]?

Arguments None

Returns 0 is returned if the signal fails the mask test.

1 is returned if the signal passes the mask test.

Examples FETCH:FSETTLING:MASK:PASS? might return 1, indicating the signal passed (did not exceed the mask limits).

FETCh:{FSETtling|PSETtling}:SETTled:FREQuency? (Query Only)

Returns the frequency at which the signal is considered settled.

Conditions Measurement views: Frequency and Phase Settling

Group Fetch commands

Syntax FETCh:{FSETtling|PSETtling}:SETTled:FREQuency?

Arguments None

Returns <value> :: <NRf> is the frequency at which the signal is settled.

Examples FETCH:FSETTLING:SETTLED:FREQUENCY? might return 2.44838E+9, indicating the input signal frequency at the point where the signal is considered settled is 2.44838 GHz.

FETCh:{FSETtling|PSETtling}:SETTled[:PASS]? (Query Only)

Returns whether the input signal is settled with the tolerance range.

Conditions Measurement views: Frequency and Phase Settling

Group Fetch commands

Syntax FETCh:{FSETtling|PSETtling}:SETTled[:PASS]?

Arguments None

Returns 0 is returned if the signal is not settled within the tolerance range.

1 is returned if the signal is settled within the tolerance range.

Examples FETCH:FSETTLING:SETTLED:PASS? might return 1, indicating the signal is settled within the tolerance range.

FETCh:{FSETtling|PSETtling}:SETTled:TIME? (Query Only)

Returns the settled time in seconds. The settled time is measured from the measurement start point.

Conditions Measurement views: Frequency and Phase Settling

Group Fetch commands

Syntax FETCh:{FSETtling|PSETtling}:SETTled:TIME?

Arguments None

Returns <value> :: <NRf> is the settled time in seconds.

Examples FETCH:FSETTLING:SETTLED:TIME? might return 299.830000E-6, indicating the settled time is 299.83 µs.

FETCh:{FSETtling|PSETtling}:SLMSd[:PASS]? (Query Only)

Returns whether the input signal is settled within the specified tolerance and the signal is settled longer than the Minimum Settled Duration ([SENSe]:{FSETtling|PSETtling}:SDURation:MINimum).

Conditions Measurement views: Frequency and Phase Settling

Group Fetch commands

Syntax FETCh:{FSETtling|PSETtling}:SLMSd[:PASS]?

Arguments None

Returns 0 is returned if the signal is not settled within the tolerance range and minimum settled duration..

1 is returned if the signal is settled within the tolerance range and minimum settled duration.

Examples FETCH:FSETTLING:SLMSD:PASS? might return 1, indicating the signal is settled within the tolerance range and the minimum settled duration.

FETCh:{FSETtling|PSETtling}:STARt:TIME? (Query Only)

Returns the start time for the measurement in seconds. The start time is measured from the start of the analysis period.

Conditions Measurement views: Frequency and Phase Settling

Group Fetch commands

Syntax FETCh:{FSETtling|PSETtling}:STARt:TIME?

Arguments None

Returns <value> :: <NRf> is the time in seconds when the measurement started.

Examples FETCH:FSETTLING:START:TIME? might return 251.4300E-6, indicating the measurement started 251.43 µs after the beginning of the analysis period.

FETCh:{FSETtling|PSETtling}:TIME? (Query Only)

Returns the settling time in seconds. The settling time is measured from the start time (see :FETCh:FSETtling:STARt:TIME).

Conditions Measurement views: Frequency and Phase Settling

Group Fetch commands

Syntax FETCh:{FSETtling|PSETtling}:TIME?

Arguments None

Returns <value> :: <NRf> is the settling time in seconds.

Examples FETCH:FSETTLING:TIME? might return 48.4000E-6, indicating the settling time is 48.4 µs.

FETCh:{FSETtling|PSETtling}:TRACe<x>:X? (Query Only)

Returns the Frequency or Phase values of the specified trace.

The parameter <x> = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions Measurement views: Frequency and Phase Settling

Group Fetch commands

Syntax FETCh:{FSETtling|PSETtling}:TRACe<x>:X?

Arguments None

Returns #<num_digit><num_byte><y(1)><y(2)>...<y(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<y(n)> is the frequency (Hz) or phase (degrees) at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples FETCH:FSETTLING:TRACE1:X? might return #574232xxxx... (74232-byte data) for the frequency values of Trace 1.

FETCh:{FSETtling|PSETtling}:TRACe<x>:XY? (Query Only)

Returns the time and frequency or phase value pairs of the specified trace.

The parameter <x> = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions Measurement views: Frequency and Phase Settling

Group Fetch commands

Syntax `FETCh:{FSETtling|PSETtling}:TRACe<x>:XY?`

Arguments None

Returns `#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>`

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<x(n)><y(n)> is the time and frequency or phase value pair at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:FSETTLING:TRACE2:XY?` might return `#574232xxxx...` (74232-byte data) for the time and frequency pairs of the Trace 2.

FETCh:{FSETtling|PSETtling}:TRACe<x>[:Y]? (Query Only)

Returns the frequency or phase values of the specified trace.

The parameter <x> = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions Measurement views: Frequency and Phase Settling

Group Fetch commands

Syntax `FETCh:{FSETtling|PSETtling}:TRACe<x>[:Y]?`

Arguments None

Returns `#<num_digit><num_byte><y(1)><y(2)>...<y(n)>`

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<y(n)> is the frequency or phase value at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:FSETTLING:TRACE1:Y?` might return `#575148xxxx...` (75148-byte data) for the frequency values of Trace 1.

FETCh:{FSETtling|PSETtling}:TRIGger:TIME? (Query Only)

Returns the time when the trigger occurred in seconds.

Conditions Measurement views: Frequency and Phase Settling

Group Fetch commands

Syntax `FETCh:{FSETtling|PSETtling}:TRIGger:TIME?`

Arguments None

Returns `<value> :: <NRf>` is the time in seconds after the measurement start point when the trigger occurred.

Examples `FETCH:FSETTLING:TRIGGER:TIME?` might return `2.255E-6`, indicating the time when the trigger occurred 255 µs after the start of the measurement.

FETCh:{FSETtling|PSETtling}:VALue? (Query Only)

Returns the settled value in Hz for Frequency Settling and in degrees for Phase Settling.

Conditions Measurement views: Frequency and Phase Settling

Group Fetch commands

Syntax `FETCh:{FSETtling|PSETtling}:VALue?`

Arguments None

Returns `<value> :: <NRf>` is the settling value in Hz for Frequency Settling and in degrees for Phase Settling.

Examples `FETCH:FSETTLING:VALUE?` might return `2.44838155E+9`, indicating the settled frequency is 2.44838 GHz.

FETCh:FSETtling:ERRor? (Query Only)

Returns the settled error in Hz in Frequency Settling. Only for Frequency Settling Time measurement. In Frequency Settling:

- When Target Reference is set to Auto, Settled Error = 0.

Conditions Measurement views: Frequency Settling

Group Fetch commands

Syntax `FETCh:FSETtling:ERRor?`

Arguments None

Returns `<value> :: <NRf>` returns the settled error in Hz.

When Target Reference is set to Auto, Settled Error = 0.

When Target Reference is set to Meas Freq:

Settled Error = Settled Frequency – (Measurement Frequency + Offset)

Examples `FETCH:FSETTLING:FTTIME?` might return `0`, indicating the Target Reference is set to Auto.

FETCh:FVTime? (Query Only)

Returns the Frequency versus Time trace data.

Conditions Measurement views: Frequency versus Time

Group Fetch commands

Syntax `FETCh:FVTime?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the frequency in Hz at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:FVTIME?` might return `#3156xxxx...` (156-byte data) for the Frequency versus Time trace.

FETCh:FVTime:MAXimum? (Query Only)

Returns the maximum value in the Frequency versus Time measurement.

Conditions Measurement views: Frequency versus Time

Group Fetch commands

Syntax `FETCh:FVTime:MAXimum?`

Related Commands FETCh:FVTime:MAXLocation?

Arguments None

Returns `<max> :: <NRf>` is the maximum frequency drift in Hz.

Examples `FETCH:FVTIME:MAXIMUM?` might return `2.625E+6`, indicating the maximum frequency drift is 2.625 MHz.

FETCh:FVTime:MAXLocation? (Query Only)

Returns the time at which the frequency drift is maximum.

Conditions Measurement views: Frequency versus Time

Group Fetch commands

Syntax `FETCh:FVTime:MAXLocation?`

Related Commands FETCh:FVTime:MAXimum?

Arguments None

Returns `<max_time> ::` `<NRf>` is the time in seconds at which the frequency drift is maximum.

Examples `FETCH:FVTIME:MAXLOCATION?` might return `25.03E-9`, indicating the frequency drift is maximum at 25.03 ns.

FETCh:FVTime:MINimum? (Query Only)

Returns the minimum value in the Frequency versus Time measurement.

Conditions Measurement views: Frequency versus Time

Group Fetch commands

Syntax `FETCh:FVTime:MINimum?`

Related Commands FETCh:FVTime:MINLocation?

Arguments None

Returns `<min> ::` `<NRf>` is the minimum frequency drift in Hz.

Examples `FETCH:FVTIME:MINIMUM?` might return `-6.618E+6`, indicating the minimum frequency drift is -6.618 MHz.

FETCh:FVTime:MINLocation? (Query Only)

Returns the time at which the frequency drift is minimum.

Conditions Measurement views: Frequency versus Time

Group Fetch commands

Syntax `FETCh:FVTime:MINLocation?`

Related Commands FETCh:FVTime:MINimum?

Arguments None

Returns `<min_time> ::` `<NRf>` is the time in seconds at which the frequency drift is minimum.

Examples `FETCH:FVTIME:MINLOCATION?` might return `450.7E-9`, indicating the frequency drift is minimum at 450.7 ns.

FETCh:FVTime:RESult? (Query Only)

Returns the Frequency versus Time measurement results.

Conditions Measurement views: Frequency versus Time

Group Fetch commands

Syntax `FETCh:FVTime:RESult?`

Arguments None

Returns `<max>,<max_time>,<min>,<min_time>`

Where
`<max> ::` `<NRf>` is the maximum frequency drift in Hz.
`<max_time> ::` `<NRf>` is the time in seconds at which the frequency drift is maximum.
`<min> ::` `<NRf>` is the minimum frequency drift in Hz.
`<min_time> ::` `<NRf>` is the time in seconds at which the frequency drift is minimum.

Examples `FETCH:FVTIME:RESULT?` might return `2.625E+6,25.03E-9,-6.618E+6,450.7E-9`, indicating the maximum frequency drift is 2.625 MHz at 25.03 ns and the minimum frequency drift is -6.618 MHz at 450.7 ns.

FETCh:IQVTime:I? (Query Only)

Returns the I versus Time trace data.

Conditions Measurement views: RF I&Q versus Time

Group Fetch commands

Syntax `FETCh:IQVTime:I?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the I level in volts at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:IQVTIME:I?` might return `#3160xxxx...` (160-byte data) for the I versus Time trace.

FETCh:IQVTime:MAXimum? (Query Only)

Returns the maximum value in the RF I&Q versus Time measurement.

Conditions Measurement views: RF I&Q versus Time

Group Fetch commands

Syntax `FETCh:IQVTime:MAXimum?`

Related Commands FETCh:IQVTime:MAXLocation?

Arguments None

Returns <max> :: <NRf> is the maximum I or Q level in volts.

Use the TRACe:IQVTime:SELect:I or TRACe:IQVTime:SELect:Q command to select the trace.

Examples FETCH:IQVTIME:MAXIMUM? might return 1.214, indicating the maximum I or Q level is 1.214 V.

FETCh:IQVTime:MAXLocation? (Query Only)

Returns the time at which the I or Q level is maximum.

Conditions Measurement views: RF I&Q versus Time

Group Fetch commands

Syntax FETCh:IQVTime:MAXLocation?

Related Commands FETCh:IQVTime:MAXimum?

Arguments None

Returns <max_time> :: <NRf> is the time in seconds at which the I or Q level is maximum.

Examples FETCH:IQVTIME:MAXLOCATION? might return 175.3E-9, indicating the I or Q level is maximum at 175.3 ns.

FETCh:IQVTime:MINimum? (Query Only)

Returns the minimum value in the RF I&Q versus Time measurement.

Conditions Measurement views: RF I&Q versus Time

Group Fetch commands

Syntax `FETCh:IQVTime:MINimum? imum`

Related Commands FETCh:IQVTime:MINLocation?

Arguments None

Returns `<min> ::` `<NRf>` is the minimum I or Q level in volts.

Use the TRACe:IQVTime:SELect:I or TRACe:IQVTime:SELect:Q command to select the trace.

Examples `FETCH:IQVTIME:MINIMUM?` might return `-370.5E-3`, indicating the minimum I or Q level is -370.5 mV.

FETCh:IQVTime:MINLocation? (Query Only)

Returns the time at which the I or Q level is minimum.

Conditions Measurement views: RF I&Q versus Time

Group Fetch commands

Syntax `FETCh:IQVTime:MINLocation?`

Related Commands FETCh:IQVTime:MINimum?

Arguments None

Returns `<min_time> ::` `<NRf>` is the time in seconds at which the I or Q level is minimum.

Examples `FETCH:IQVTIME:MINLOCATION?` might return `450.7E-9`, indicating the I or Q level is minimum at 450.7 ns.

FETCh:IQVTime:Q? (Query Only)

Returns the Q versus Time trace data.

Conditions Measurement views: RF I&Q versus Time

Group Fetch commands

Syntax `FETCh:IQVTime:Q?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the Q level in volts at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:IQVTIME:Q?` might return `#3160xxxx...` (160-byte data) for the Q versus Time trace.

FETCh:IQVTime:RESult? (Query Only)

Returns the RF I&Q versus Time measurement results.

Conditions Measurement views: RF I&Q versus Time

Group Fetch commands

Syntax `FETCh:IQVTime:RESult?`

Arguments None

Returns `<max>,<max_time>,<min>,<min_time>`

Where
`<max> ::` `<NRf>` is the maximum I or Q level in volts.

<max_time> :: <NRf> is the time in seconds at which the I or Q level is maximum.
<min> :: <NRf> is the minimum I or Q level in volts.
<min_time> :: <NRf> is the time in seconds at which the I or Q level is minimum.

Use the TRACe:IQVTime:SELect:I or TRACe:IQVTime:SELect:Q command to select the trace.

Examples FETCH:IQVTIME:RESULT? might return 1.214,175.3E-9,-370.5E-3,450.7E-9, indicating the maximum I or Q level is 1.214 V at 175.3 ns and the minimum I or Q level is -370.5 mV at 450.7 ns.

FETCh:LTE:ACLR:MHITs:COUNt? (Query Only)

Returns the number of rows in the results table of the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Fetch commands

Syntax FETCh:LTE:ACLR:MHITs:COUNt?

Returns <NR1>

Examples FETCH:LTE:ACLR:MHITS:COUNT? might return 7, indicating there are seven rows in the results table.

FETCh:LTE:ACLR:MHITs<x>:BANDwidth? (Query Only)

Returns the bandwidth value of the channel (row) of the LTE ACLR display.

Conditions Measurement view: LTE ACLR

The parameter <x> is the row number in the LTE ACLR display results table.

Group Fetch commands

Syntax FETCh:LTE:ACLR:MHITs<x>:BANDwidth?

Returns <NR3>

Examples FETCh:LTE:ACLR:MHITS1:BANDWIDTH? might return 4.5150000000E+6, which indicates a bandwidth for Channel 1 of 4.515 MHz.

FETCh:LTE:ACLR:MHITs<x>:CHANnel:NAME? (Query Only)

Returns the channel name of the specified index in the results table of the LTE ACLR display. The minimum index will be 1 and the maximum index will be the count of the channels in the results table.

Conditions Measurement view: LTE ACLR

The parameter <x> is a positive integer and represents the row number in the LTE ACLR display results table.

Group Fetch commands

Syntax FETCh:LTE:ACLR:MHITs<x>:CHANnel:NAME?

Returns <string>

Examples FETCH:LTE:ACLR:MHITS2:CHANNEL:NAME? might return “B”, which is the name of channel 2.

FETCh:LTE:ACLR:MHITs<x>:FREQuency:OFFSet? (Query Only)

Returns the frequency offset of the specified channel (row) of the LTE ACLR display.

Conditions Measurement view: LTE ACLR

The parameter <x> is a positive integer and represents the row number in the LTE ACLR display results table.

Group Fetch commands

Syntax FETCh:LTE:ACLR:MHITs<x>:FREQuency:OFFSet?

Returns <NR3>

Examples FETCh:LTE:ACLR:MHITS1:FREQUENCY:OFFSET? might return 5.0000000000E+6 which indicates the frequency offset of Channel 1 is 5 MHz.

FETCh:LTE:ACLR:MHITs<x>:INTeg:LOWEr:ABSolute? (Query Only)

Returns the lower absolute value of the channel (row) of the LTE ACLR display.

Conditions Measurement view: LTE ACLR

The parameter <x> is the row number in the LTE ACLR display results table.

Group Fetch commands

Syntax FETCh:LTE:ACLR:MHITs<x>:INTeg:LOWEr:ABSolute?

Returns <NRf>

Examples FETCh:LTE:ACLR:MHITS1:INTEG:LOWER:ABSOLUTE? might return -53.3921980303, which indicates the lower absolute value of Channel 1 is -53.3921980303 dBm.

NOTE. *The actual units depends on the choice made from the units tab of the Analysis control panel.*

FETCh:LTE:ACLR:MHITs<x>:INTeg:LOWEr:RELative? (Query Only)

Returns the lower relative value of the channel (row) of the LTE ACLR display.

Conditions Measurement view: LTE ACLR

The parameter <x> is the row number in the LTE ACLR display results table.

Group Fetch commands

Syntax `FETCh:LTE:ACLR:MHITs<x>:INTeg:LOWEr:RELative?`

Returns <NRf>

Examples `FETCh:LTE:ACLR:MHITS1:INTEG:LOWER:RELATIVE?` might return -53.3921980303 which indicates lower relative value of Channel 1 is -53.3921980303 dB.

FETCh:LTE:ACLR:MHITs<x>:INTeg:UPPEr:ABSolute? (Query Only)

Returns the upper absolute value of the channel (row) of the LTE ACLR display.

Conditions Measurement view: LTE ACLR

The parameter <x> is a positive integer and represents the row number in the LTE ACLR display results table.

Group Fetch commands

Syntax `FETCh:LTE:ACLR:MHITs<x>:INTeg:UPPEr:ABSolute?`

Returns <NRf>

Examples `FETCH:LTE:ACLR:MHITS1:INTEG:UPPER:ABSOLUTE?` might return –53.3921980303 which indicates upper absolute value of Channel 1 is -53.3921980303 dBm.

NOTE. *The actual units depends on the choice made from the units tab of the Analysis control panel.*

FETCh:LTE:ACLR:MHITs<x>:INTeg:UPPEr:RELative? (Query Only)

Returns the upper relative value of the channel (row) of the LTE ACLR display.

Conditions Measurement view: LTE ACLR

The parameter <x> is the row number in the LTE ACLR display results table.

Group Fetch commands

Syntax `FETCh:LTE:ACLR:MHITs<x>:INTeg:UPPEr:RELative?`

Returns <NRf>

Examples `READ:LTE:ACLR:MHITS1:INTEG:UPPER:RELATIVE?` might return -53.3921980303 which indicates upper relative value of Channel 1 is -53.3921980303 dB.

FETCh:LTE:ACLR:REFerence:POWer? (Query Only)

Returns the reference power level in the LTE ACLR measurement.

Conditions Measurement view: LTE ACLR

Group Fetch commands

Syntax `FETCh:LTE:ACLR:REFerence:POWer?`

Returns <NRf>

Examples `FETCH:LTE:ACLR:REFERENCE:POWER?` might return –16.92, indicating that the reference power level is –16.92 dBm.

FETCh:LTE:ACLR:RESUlts:STATus? (Query Only)

Returns the pass or fail status of the LTE ACLR display measurement.

Conditions Measurement view: LTE ACLR

Group Fetch commands

Syntax `FETCh:LTE:ACLR:RESUlts:STATus?`

Returns 0: means measurement status is FAIL.

1: means measurement status is PASS.

Examples `FETCH:LTE:ACLR:RESULTS:STATUS?` might return 1, indicating that the LTE ACLR measurement status is PASS.

FETCh:LTE:ACLR:SPECtrum:X? (Query Only)

Returns the frequencies of the spectrum trace in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Fetch commands

Syntax `FETCh:LTE:ACLR:SPECtrum:X?`

Returns #<num_digit><num_byte><x(1)><x(2)>...<x(n)>

Where,

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<x(n)> is the frequency (Hz) at the n^{th} point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:LTE:ACLR:SPECTRUM:X?` might return #516020xxxx... (16020-byte data) for the frequencies of the spectrum trace.

FETCh:LTE:ACLR:SPECtrum:XY? (Query Only)

Returns the frequency and amplitude pairs of the spectrum trace in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Fetch commands

Syntax `FETCh:LTE:ACLR:SPECtrum:XY?`

Returns #<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>

Where,

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<x(n)><y(n)> is the frequency (Hz) and amplitude (dBm) pair at the n^{th} point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:LTE:ACLR:SPECTRUM:XY?` might return #516020xxxx... (16020-byte data) for the frequency and amplitude pairs of the spectrum trace.

FETCh:LTE:ACLR:SPECtrum:Y? (Query Only)

Returns the amplitude of the spectrum trace in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Fetch commands

Syntax `FETCh:LTE:ACLR:SPECtrum:Y?`

Returns #<num_digit><num_byte><y(1)><y(2)>...<y(n)>

Where,

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<y(n)> is the amplitude (dB) at the n^{th} point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:LTE:ACLR:SPECTRUM:Y?` might return #516020xxxx... (16020-byte data) for the amplitude of the spectrum trace.

FETCh:LTE:CHSPectrum:OBW? (Query Only)

Returns the Occupied Bandwidth (OBW) in Hz of the LTE Channel Spectrum measurement view.

Conditions Measurement view: LTE Channel Spectrum

Group Fetch commands

Syntax `FETCh:LTE:CHSPectrum:OBW?`

Returns <NR3>

Examples `FETCH:LTE:CHSPECTRUM:OBW?` might return 1.0851851250E+6, indicating the Occupied Bandwidth is 1.09 MHz.

FETCh:LTE:CHSPectrum:POWer:CHANnel? (Query Only)

Returns the Channel Power value in LTE Channel Spectrum display. The result will be in dBm (by default) or in the units chosen from the Units tab of the Analysis Control panel.

Conditions Measurement view: LTE Channel Spectrum

Group Fetch commands

Syntax `FETCh:LTE:CHSPectrum:POWer:CHANnel?`

Returns <NRf>

Examples `FETCH:LTE:CHSPECTRUM:POWER:CHANNEL?` might return –14.9248560147, indicating that the Channel Power is -14.92 dBm.

FETCh:LTE:CHSPectrum:SPECtrum? (Query Only)

Returns spectrum trace data of the LTE Channel Spectrum measurement.

Conditions Measurement view: LTE Channel Spectrum

Group Fetch commands

Syntax `FETCh:LTE:CHSPectrum:SPECtrum?`

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of the data that follow.

<data(n)> is the amplitude in dBm at the n^{th} data poin, 4-byte little endian floating point format specified in IEEE 488.2.

Examples `FETCH:LTE:CHSPECTRUM:SPECTRUM?` might return #43204 xxxx... (3204-byte data) for the spectrum trace data of the LTE Channel Spectrum measurement.

FETCh:LTE:CONSte:CELL:ID? (Query Only)

Queries the Cell ID value LTE Constellation display.

Conditions Measurement view: LTE Constellation

Group Fetch commands

Syntax `FETCh:LTE:CONSte:CELL:ID?`

Returns <NR1>

Examples `FETCH:LTE:CONSTE:CELL:ID? ?` might return 112, indicating CELLID measurement number is 112.

FETCh:LTE:CONSte:FREQuency:ERRor? (Query Only)

Queries Frequency Error value in Hz LTE Constellation display.

Conditions Measurement view: LTE Constellation

Group Fetch commands

Syntax `FETCh:LTE:CONSte:FREQuency:ERRor?`

Returns <NRf>

Examples FETCH:LTE:CONSTE:FREQUENCY:ERROR? might return -71.9780578613, indicating frequency error value is -71.9780578613 Hz.

FETCh:LTE:CONSte:GROUp:ID? (Query Only)

Queries the Group ID value LTE Constellation display.

Conditions Measurement view: LTE Constellation

Group Fetch commands

Syntax FETCh:LTE:CONSte:GROUp:ID?

Returns <NR1>

Examples FETCH:LTE:CONSTE:GROUP:ID? ? might return 160, indicating Group ID measurement number is 160.

FETCh:LTE:CONSte:SECTor:ID? (Query Only)

Queries the Sector ID value for the LTE Constellation display.

Conditions Measurement view: LTE Constellation

Group Fetch commands

Syntax FETCh:LTE:CONSte:SECTor:ID?

Returns <NRf>

Examples FETCH:LTE:CONSTE:SECTOR:ID? might return 2, indicating the Sector ID value is 2.

FETCh:LTE:PVTime:OFFSlot:POWer? (Query Only)

Returns the TOff power measurement value in dBm/MHz for the LTE Power vs Time display.

Conditions Measurement view: LTE Power vs Time

Group Fetch commands

Syntax `FETCh:LTE:PVTime:OFFSlot:POWer?`

Returns <NRf>

Examples `FETCH:LTE:PVTIME:OFFSLOT:POWER?` might return -76.11514587403, indicating the offslot power is -76.11514 dBm/MHz.

FETCh:LTE:PVTime:RESUlts:STATus

Returns the pass or fail status for the LTE Power vs Time display.

Conditions Measurement view: LTE Power vs Time

Group Fetch commands

Syntax `FETCh:LTE:PVTime:RESUlts:STATus`

Returns 0: means measurement status is FAIL.

1: means measurement status is PASS.

Examples `FETCH:LTE:PVTIME:RESULTS:STATUS` might return 1, indicating the measurement status is PASS.

FETCh:LTE:PVTime:TRACe:X? (Query Only)

Returns the horizontal values (time in seconds) for the LTE Power vs. Time trace.

Conditions Measurement view: LTE Power vs Time

Group Fetch commands

Syntax `FETCh:LTE:PVTime:TRACe:X?`

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where,

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the n^{th} point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:LTE:PVTIME:TRACE:X?` might return #43204 (3204-byte data), which represents the time in seconds (horizontal values).

FETCh:LTE:PVTime:TRACe:XY? (Query Only)

Returns the horizontal value (time in seconds) and vertical value (power) for the LTE Power vs. Time trace.

Conditions Measurement view: LTE Power vs Time

Group Fetch commands

Syntax `FETCh:LTE:PVTime:TRACe:XY?`

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where,

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the n^{th} point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:LTE:PVTIME:TRACE:XY?` might return #43204 (3204-byte data), which represents the horizontal value (time in seconds) and vertical value (power) pair at the n^{th} data point.

FETCh:LTE:PVTime:TRACe:Y? (Query Only)

Returns the vertical values (power) for the LTE Power vs. Time trace.

Conditions Measurement view: LTE Power vs Time

Group Fetch commands

Syntax `FETCh:LTE:PVTime:TRACe:Y?`

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where,

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the n^{th} point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:LTE:PVTIME:TRACE:Y?` might return #43204 (3204-byte data), which represents the vertical values (power).

FETCh:MCPower:ADJacent:CHANnels? (Query Only)

Returns the power of adjacent channels in order of increasing frequency.

Conditions Measurement views: MCPR

Group Fetch commands

Syntax `FETCh:MCPower:ADJacent:CHANnels?`

Arguments None

Returns

<acpr_lower(n)>,...<acpr_lower(2)>,<acpr_lower(1)>,
<acpr_upper(1)>,<acpr_upper(2)>,...<acpr_upper(n)>

Where
<acpr_lower(n)> is the ACPR for the lower channel #n in dB.
<acpr_upper(n)> is the ACPR for the upper channel #n in dB.

To add a pair of upper and lower adjacent channels, use the [SENSe]:MCPower:CHANnel:ADJacent:ADD command.

Examples

FETCH:MCPOWER:ADJACENT:CHANNELS? might return -4.420,-4.847,-4.316,-4.225, indicating
(ACPR for the lower channel 2) = -4.420 dB,
(ACPR for the lower channel 1) = -4.847 dB,
(ACPR for the upper channel 1) = -4.316 dB, and
(ACPR for the upper channel 2) = -4.225 dB.

FETCh:MCPower:CHANnel:POWer? (Query Only)

Returns the reference power in the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Fetch commands

Syntax

FETCh:MCPower:CHANnel:POWer?

Arguments

None

Returns

<ref_power>:<NRf> is the reference power in dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

To select the power reference, use the [SENSe]:MCPower:RCHannels? commands.

Examples

FETCH:MCPOWER:CHANNEL:POWER? might return 4.227, indicating that the reference power is 4.227 dBm.

FETCh:MCPower:MAIN:CHANnels? (Query Only)

Returns the power of main channels in order of increasing frequency.

Conditions Measurement views: MCPR

Group Fetch commands

Syntax `FETCh:MCPower:MAIN:CHANnels?`

Arguments None

Returns `<power_main(1)>,<power_main(2)>,...<power_main(n)>`

Where
`<power_main(n)>` is the power of main channel #n in dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

To specify the main channels, use the [SENSe]:MCPower:CHANnel:MAIN commands.

Examples `FETCH:MCPOWER:MAIN:CHANNELS?` might return `-2.420,-2.847,-2.316,-2.225`, indicating
(power of the main channel 1) = -2.420 dBm,
(power of the main channel 2) = -2.847 dBm,
(power of the main channel 3) = -2.316 dBm, and
(power of the main channel 4) = -2.225 dBm.

FETCh:MCPower:SPECtrum? (Query Only)

Returns spectrum trace data of the MCPR measurement.

Conditions Measurement views: MCPR

Group Fetch commands

Syntax `FETCh:MCPower:SPECtrum?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.
<data(n)> is the amplitude in dBm at the nth data point,
4-byte little endian floating-point format specified in IEEE 488.2.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples FETCH:MCPOWER:SPECTRUM? might return #43204xxxx... (3204-byte data) for the spectrum trace data of the MCPR measurement.

FETCh:MERRor:FERRor? (Query Only)

Returns the frequency error in the Magnitude error versus Time measurement.

Conditions Measurement views: Magnitude error versus Time

Group Fetch commands

Syntax FETCh:MERRor:FERRor?

Arguments None

Returns <freq_error> :: <NRf> is the frequency error in Hz.

Examples FETCH:MERROR:FERROR? might return -10.7E+3, indicating the frequency error is -10.7 kHz.

FETCh:MERRor:PEAK? (Query Only)

Returns the peak value in the Magnitude error versus Time measurement.

Conditions Measurement views: Magnitude error versus Time

Group Fetch commands

Syntax FETCh:MERRor:PEAK?

Related Commands FETCh:MERRor:PINDex?

Arguments None

Returns <peak> :: <NRf> is the peak magnitude error in percent (%).

Examples FETCH:MERROR:PEAK? might return 1.57, indicating the peak magnitude error is 1.57%.

FETCh:MERRor:PINDex? (Query Only)

Returns the time at the magnitude error peak.

Conditions Measurement views: Magnitude error versus Time

Group Fetch commands

Syntax FETCh:MERRor:PINDex?

Related Commands FETCh:MERRor:PEAK?

Arguments None

Returns <peak_time> :: <NRf> is the time at the magnitude error peak in symbol number. The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples FETCH:MERROR:PINDEX? might return 68.000, indicating that the magnitude error peak is at symbol #68.

FETCh:MERRor:RMS? (Query Only)

Returns the RMS (Root-Mean-Square) value in the Magnitude error versus Time measurement.

Conditions Measurement views: Magnitude error versus Time

Group Fetch commands

Syntax FETCh:MERRor:RMS?

Arguments None

Returns <rms> :: <NRf> is the RMS magnitude error in percent (%).

Examples FETCH:MERROR:RMS? might return 0.382, indicating the magnitude error is 0.382% RMS.

FETCh:MERRor:TRACe? (Query Only)

Returns the Magnitude error versus Time trace data.

Conditions Measurement views: Magnitude error versus Time

Group Fetch commands

Syntax FETCh:MERRor:TRACe?

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the magnitude error in percent (%) at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples FETCH:MERROR:TRACE? might return #42036xxxx... (2036-byte data) for the Magnitude error versus Time trace.

FETCh:NOISe:GAIN:TRACe<x>:RESult? (Query Only)

This query returns the gain value in mdB of the specified trace in the Gain display.

Conditions Measurement view: Gain

The parameter <x> = 1 to 3.

Group Fetch commands

Syntax `FETCh:NOISe:GAIN:TRACe<x>:RESult?`

Returns The gain value in mdB of the specified trace.

Examples `FETCH:NOISE:GAIN:TRACE1:RESULT?` might return -88.7, indicating that the gain is -88.7 mdB

FETCh:NOISe:TABLe:VALue? (Query Only)

Returns Noise Table data. A total of 7 values per frequency (28 bytes per frequency) is returned. The number of frequencies returned depends on the settings.

Conditions Measurement view: Noise Table

Group Fetch commands

Syntax `FETCh:NOISe:TABLe:VALue?`

Returns Returns an array of floats (size of 4 bytes each) defined as:

- First frequency gain, noise figure, noise factor, y factor, noise temperature, power hot, power cold
- Second frequency gain, noise figure, noise factor, y factor, noise temperature, power hot, power cold

Examples `FETCH:NOISE:TABLE:VALUE?` might return #3308, indicating that the

FETCh:OBWidth:BOBW:XDBBandwidth? (Query Only)

Returns the difference between the higher and lower frequency points corresponding to the value where it is X dB less from the value at the center frequency.

Set the measurement direction using the appropriate inward or outward command before issuing this command.

Conditions Measurement views: Bluetooth 20 dB Bandwith

Group Fetch commands

Syntax `FETCh:OBWidth:BOBW:XDBBandwidth?`

Related Commands FETCh:OBWidth:BOBW:XDBMbandwidth:IN:BANDwidth?, FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:BANDwidth?

Arguments None

Returns <value> ::= <NRf> the X dBm bandwidth in Hz.

Examples `FETCH:OBWIDTH:BOBW:XDBBANDWIDTH?` might return 89.12000E+3 indicating the difference between the higher and lower frequency points is 89.12 kHz.

FETCh:OBWidth:BOBW:XDBMbandwidth:IN:BANDwidth? (Query Only)

Returns the difference between the higher and lower frequency points corresponding to the value which it is the X dBm value measured inwards (coming from the ends to the center frequency).

Conditions Measurement views: Bluetooth 20 dB Bandwith

Group Fetch commands

Syntax `FETCh:OBWidth:BOBW:XDBMbandwidth:IN:BANDwidth?`

Related Commands FETCh:OBWidth:BOBW:XDBBandwidth?, FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:BANDwidth?

Arguments None

Returns <value> ::= <NRf> the input X dBm value in Hz.

Examples `FETCH:OBWIDTH:BOBW:XDBMBANDWIDTH:IN:BANDWIDTH?` might return 1.2518400000E+6 indicating that the bandwidth corresponding to the input XdBm vaue is 1.2518 MHz measured in the inward direction.

FETCh:OBWidth:BOBW:XDBMbandwidth:IN:LEFT:FREQuency? (Query Only)

Returns the frequency corresponding to the X dBm value to the left of the center frequency measured in the inward direction.

Conditions Measurement views: Bluetooth 20 dB Bandwith

Group Fetch commands

Syntax `FETCh:OBWidth:BOBW:XDBMbandwidth:IN:LEFT:FREQuency?`

Related Commands FETCh:OBWidth:BOBW:XDBMbandwidth:IN:RIGHt:FREQuency?

Arguments None

Returns <value> ::= <NRf> the left frequency in Hz.

Examples `FETCH:OBWIDTH:BOBW:XDBMBANDWIDTH:IN:LEFT:FREQUENCY?` might return 2.401E+9 indicating that the frequency corresponding to the X dbm value to the left of the center frequency is 2.401GHz.

FETCh:OBWidth:BOBW:XDBMbandwidth:IN:RIGHt:FREQuency? (Query Only)

Returns the frequency corresponding to the X dBm value to the right of the center frequency measured in the inward direction.

Conditions Measurement views: Bluetooth 20 dB Bandwith

Group Fetch commands

Syntax `FETCh:OBWidth:BOBW:XDBMbandwidth:IN:RIGHt:FREQuency?`

Related Commands FETCh:OBWidth:BOBW:XDBMbandwidth:IN:LEFT:FREQuency?

Arguments None

Returns <value> ::= <NRf> the right frequency in Hz.

Examples `FETCH:OBWIDTH:BOBW:XDBMBANDWIDTH:IN:RIGHT:FREQUENCY?` might return 2.445E+9 indicating that the frequency corresponding to the XdBm value to the right of the center the frequency is 2.445 GHz.

FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:BANDwidth? (Query Only)

Returns the difference between the higher and lower frequency points corresponding to the value which it is the X dBm value measured outwards (from the center frequency to the ends).

Conditions Measurement views: Bluetooth 20 dB Bandwith

Group Fetch commands

Syntax `FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:BANDwidth?`

Related Commands FETCh:OBWidth:BOBW:XDBBandwidth?, FETCh:OBWidth:BOBW:XDBMbandwidth:IN:BANDwidth?

Arguments None

Returns <value> ::= <NRf> the input X dBm value in Hz.

Examples `FETCH:OBWIDTH:BOBW:XDBMBANDWIDTH:OUT:BANDWIDTH?` might return 1.2518400000E+6 indicating that the bandwidth corresponding to the output XdBm vaue is 1.2518 MHz measured in the outward direction.

FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:LEFT:FREQuency? (Query Only)

Returns the frequency corresponding to the X dBm value to the left of the center frequency measured in the outward direction.

Conditions Measurement views: Bluetooth 20 dB Bandwith

Group Fetch commands

Syntax `FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:LEFT:FREQuency?`

Related Commands FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:RIGHt:FREQuency?

Arguments None

Returns <value> ::= <NRf> the left frequency in Hz.

Examples `FETCH:OBWIDTH:BOBW:XDBMBANDWIDTH:OUT:LEFT:FREQUENCY?` might return 2.401E+9 indicating that the frequency corresponding to the X dbm value to the left of the center frequency is 2.401GHz.

FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:RIGHt:FREQuency? (Query Only)

Returns the frequency corresponding to the X dBm value to the right of the center frequency measured in the outward direction.

Conditions Measurement views: Bluetooth 20 dB Bandwith

Group Fetch commands

Syntax `FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:RIGHt:FREQuency?`

Related Commands FETCh:OBWidth:BOBW:XDBMbandwidth:OUT:LEFT:FREQuency?

Arguments None

Returns <value> ::= <NRf> the Right frequency in Hz.

Examples `FETCH:OBWIDTH:BOBW:XDBMBANDWIDTH:OUT:RIGHT:FREQUENCY?` might return 2.445E+9 indicating that the frequency corresponding to the XdBm value to the right of the center the frequency is 2.445 GHz.

FETCh:OBWidth:FREQuency:ERRor? (Query Only)

Returns the frequency error in the Occupied Bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Fetch commands

Syntax `FETCh:OBWidth:FREQuency:ERRor?`

Arguments None

Returns `<freq_error> ::` `<NRf>` is the frequency error in Hz.

Examples `FETCH:OBWIDTH:FREQUENCY:ERROR?` might return `-10.7E+3`, indicating the frequency error is -10.7 kHz.

FETCh:OBWidth:OBWidth:BANDwidth? (Query Only)

Returns the occupied bandwidth in the Occupied Bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Fetch commands

Syntax `FETCh:OBWidth:OBWidth:BANDwidth?`

Arguments None

Returns `<OBW> ::` `<NRf>` is the occupied bandwidth in Hz.

Examples `FETCH:OBWIDTH:OBWIDTH:BANDWIDTH?` might return `4.0E+6`, indicating the occupied bandwidth is 4 MHz.

FETCh:OBWidth:OBWidth:LEFT:FREQuency? (Query Only)

Returns the left (lower) frequency of the occupied bandwidth.

Conditions Measurement views: Occupied Bandwidth

Group Fetch commands

Syntax `FETCh:OBWidth:OBWidth:LEFT:FREQuency?`

Related Commands FETCh:OBWidth:OBWidth:RIGHt:FREQuency?

Arguments None

Returns `<OBW_left_freq> ::` `<NRf>` is the left frequency in Hz.

Examples `FETCH:OBWIDTH:OBWIDTH:LEFT:FREQUENCY?` might return `1.498E+9`, indicating the left frequency is 1.498 GHz.

FETCh:OBWidth:OBWidth:LEFT:LEVel? (Query Only)

Returns the level at the left frequency of the occupied bandwidth.

Conditions Measurement views: Occupied Bandwidth

Group Fetch commands

Syntax `FETCh:OBWidth:OBWidth:LEFT:LEVel?`

Related Commands FETCh:OBWidth:OBWidth:RIGHt:LEVel?

Arguments None

Returns `<OBW_left_level> ::` `<NRf>` is the level at the left frequency in dB.

Examples `FETCH:OBWIDTH:OBWIDTH:LEFT:LEVEL?` might return `-23.5`, indicating the level at the left frequency is -23.5 dB.

FETCh:OBWidth:OBWidth:POWer? (Query Only)

Returns the reference power in the Occupied Bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Fetch commands

Syntax `FETCh:OBWidth:OBWidth:POWer?`

Arguments None

Returns `<OBW_ref_power> ::` `<NRf>` is the reference power in dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `FETCH:OBWIDTH:OBWIDTH:POWER?` might return `-10.0`, indicating the reference power is -10 dBm.

FETCh:OBWidth:OBWidth:RIGHt:FREQuency? (Query Only)

Returns the right (higher) frequency of the occupied bandwidth.

Conditions Measurement views: Occupied Bandwidth

Group Fetch commands

Syntax `FETCh:OBWidth:OBWidth:RIGHt:FREQuency?`

Related Commands FETCh:OBWidth:OBWidth:LEFT:FREQuency?

Arguments None

Returns `<OBW_right_freq> ::` `<NRf>` is the right frequency in Hz.

Examples `FETCH:OBWIDTH:OBWIDTH:RIGHT:FREQUENCY?` might return `1.502E+9`, indicating the right frequency is 1.502 GHz.

FETCh:OBWidth:OBWidth:RIGHt:LEVel? (Query Only)

Returns the level at the right frequency of the occupied bandwidth.

Conditions Measurement views: Occupied Bandwidth

Group Fetch commands

Syntax `FETCh:OBWidth:OBWidth:RIGHt:LEVel?`

Related Commands FETCh:OBWidth:OBWidth:LEFT:LEVel?

Arguments None

Returns `<OBW_right_level> ::` `<NRf>` is the level at the right frequency in dB.

Examples `FETCH:OBWIDTH:OBWIDTH:RIGHT:LEVEL?` might return `-23.5`, indicating the level at the right frequency is -23.5 dB.

FETCh:OBWidth:SPECtrum? (Query Only)

Returns spectrum trace data of the Occupied Bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Fetch commands

Syntax `FETCh:OBWidth:SPECtrum?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where

<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the amplitude in dBm at the nth data point,
4-byte little endian floating-point format specified in IEEE 488.2.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples FETCH:OBWIDTH:SPECTRUM? might return #43204xxxx... (3204-byte data) for the spectrum trace data of the Occupied Bandwidth measurement.

FETCh:OBWidth:XDBBandwidth:BANDwidth? (Query Only)

Returns the x dB bandwidth in the Occupied Bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Fetch commands

Syntax FETCh:OBWidth:XDBBandwidth:BANDwidth?

Arguments None

Returns <xdBBW> :: <NRf> is the x dB bandwidth in Hz.

Examples FETCH:OBWIDTH:XDBBANDWIDTH:BANDWIDTH? might return 2.0E+6, indicating the x dB bandwidth is 2 MHz.

FETCh:OBWidth:XDBBandwidth:LEFT:FREQuency? (Query Only)

Returns the left (lower) frequency of the x dB bandwidth.

Conditions Measurement views: Occupied Bandwidth

Group Fetch commands

Syntax FETCh:OBWidth:XDBBandwidth:LEFT:FREQuency?

Related Commands FETCh:OBWidth:XDBBandwidth:RIGHt:FREQuency?

Arguments None

Returns `<xdBBW_left_freq> ::` `<NRf>` is the left frequency in Hz.

Examples `FETCH:OBWIDTH:XDBBANDWIDTH:LEFT:FREQUENCY?` might return `1.498E+9`, indicating the left frequency is 1.498 GHz.

FETCh:OBWidth:XDBBandwidth:LEFT:LEVel? (Query Only)

Returns the level at the left frequency of the x dB bandwidth.

Conditions Measurement views: Occupied Bandwidth

Group Fetch commands

Syntax `FETCh:OBWidth:XDBBandwidth:LEFT:LEVel?`

Related Commands FETCh:OBWidth:XDBBandwidth:RIGHt:LEVel?

Arguments None

Returns `<xdBBW_left_level> ::` `<NRf>` is the level at the left frequency in dB.

Examples `FETCH:OBWIDTH:XDBBANDWIDTH:LEFT:LEVEL?` might return `-23.5`, indicating the level at the left frequency is -23.5 dB.

FETCh:OBWidth:XDBBandwidth:POWer? (Query Only)

Returns the reference power in the x dB bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Fetch commands

Syntax `FETCh:OBWidth:XDBBandwidth:POWer?`

Arguments None

Returns <xdBBW_ref_power> :: <NRf> is the reference power in dBm. The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples FETCH:OBWIDTH:XDBBANDWIDTH:POWER? might return -10.0, indicating the reference power is -10 dBm.

FETCh:OBWidth:XDBBandwidth:RIGHt:FREQuency? (Query Only)

Returns the right (higher) frequency of the x dB bandwidth.

Conditions Measurement views: Occupied Bandwidth

Group Fetch commands

Syntax FETCh:OBWidth:XDBBandwidth:RIGHt:FREQuency?

Related Commands FETCh:OBWidth:XDBBandwidth:LEFT:FREQuency?

Arguments None

Returns <xdBBW_right_freq> :: <NRf> is the right frequency in Hz.

Examples FETCH:OBWIDTH:XDBBANDWIDTH:RIGHT:FREQUENCY? might return 1.502E+9, indicating the right frequency is 1.502 GHz.

FETCh:OBWidth:XDBBandwidth:RIGHt:LEVel? (Query Only)

Returns the level at the right frequency of the x dB bandwidth.

Conditions Measurement views: Occupied Bandwidth

Group Fetch commands

Syntax FETCh:OBWidth:XDBBandwidth:RIGHt:LEVel?

Related Commands FETCh:OBWidth:XDBBandwidth:LEFT:LEVel?

Arguments None

Returns <xdBBW_right_level> :: <NRf> is the level at the right frequency in dB.

Examples FETCH:OBWIDTH:XDBBANDWIDTH:RIGHT:LEVEL? might return -23.5, indicating the level at the right frequency is -23.5 dB.

FETCh:OFDM:APOWer? (Query Only)

Returns the average power in the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax FETCh:OFDM:APOWer?

Arguments None

Returns <apower>::<NRf> the average power in dB.

Examples FETCH:OFDM:APOWER? might return –23.4584459235 indicating the average power is –23.46 dB.

FETCh:OFDM:APOWer:PEAK? (Query Only)

Returns the peak-to-average power in the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax FETCh:OFDM:APOWer:PEAK?

Arguments None

Returns <apower_peak>::<NRf> is the peak-to-average power in dBm.

Examples `FETCH:OFDM:APOWER:PEAK?` might return `10.4140096289` indicating that peak-to-average power is 10.41 dBm.

FETCh:OFDM:CONSte:MAGNitude? (Query Only)

Returns the constellation magnitude data for the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:CONSte:MAGNitude?`

Related Commands FETCh:OFDM:CONSte:PHASe?
FETCh:OFDM:CONSte:TYPE?
FETCh:OFDM:CONSte:VALue?

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the data value at the n^{th} data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:OFDM:CONSTE:MAGNITUDE?` might return `#510400xxxx...` (10400-byte data) for the measurement.

FETCh:OFDM:CONSte:PHASe? (Query Only)

Returns the constellation phase values of the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:CONSte:PHASe?`

Related Commands FETCh:OFDM:CONSte:MAGNitude?
FETCh:OFDM:CONSte:TYPE?
FETCh:OFDM:CONSte:VALue?

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the data value at the n^{th} data point in 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:OFDM:CONSTE:PHASE?` might return `#510400xxxx...` (10400-byte data) for the measurement.

FETCh:OFDM:CONSte:TYPE? (Query Only)

Returns the constellation context value of the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:CONSte:TYPE?`

Related Commands FETCh:OFDM:CONSte:MAGNitude?
FETCh:OFDM:CONSte:PHASe?
FETCh:OFDM:CONSte:VALue?

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the data value at the nth data point in 16-bit little endian integers. The integers must be translated to the proper context as listed below.

Integer value	Type (context)
0	Pilot
1	Data
2	Unused or Null

Examples `FETCH:OFDM:CONSTE:TYPE?` might return `#41352` (1352-byte data) for the measurement. The actual data must be decoded to the context type using the table above.

FETCh:OFDM:CONSte:VALue? (Query Only)

Returns the constellation value (decoded symbol) of the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:CONSte:VALue?`

Related Commands FETCh:OFDM:CONSte:MAGNitude?
FETCh:OFDM:CONSte:PHASe?
FETCh:OFDM:CONSte:TYPE?

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the data value at the nth data point in 16-bit little endian integers.

Examples `FETCH:OFDM:CONSTE:VALUE?` might return `#41352` indicating 1352 bytes of data for the measurement.

FETCh:OFDM:CPE? (Query Only)

Returns the Common Pilot Error magnitude for the OFDM measurement

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:CPE?`

Arguments None

Returns <NRf> the RMS magnitude error in percent.

Examples `FETCH:OFDM:CPE?` might return `3.7868041505` indicating that the CPE is 3.787%.

FETCh:OFDM:CRESponse:MAGNitude? (Query Only)

Returns the channel response magnitude data for the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:CRESponse:MAGNitude?`

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:OFDM:CRESPONSE:MAGNITUDE?` might return `#3804xxxx. . .` (804 byte data) for the measurement.

FETCh:OFDM:CRESponse:PHASe? (Query Only)

Returns the channel response phase data for the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:CRESponse:PHASe?`

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:OFDM:CRESPONSE:PHASE?` might return `#3804xxxx. . .` (804 byte data) for the measurement.

FETCh:OFDM:EVM:PEAK:DECibel:ALL? (Query Only)

Returns the peak EVM data for all subcarriers in the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:EVM:PEAK:DECibel:ALL?`

Arguments None

Returns <NRf> data in dB.

Examples FETCH:OFDM:EVM:PEAK:DECIBEL:ALL? might return –20.1872549032 indicating the data is –20.19 dB for the measurement.

FETCh:OFDM:EVM:PEAK:DECibel:DATA? (Query Only)

Returns the peak EVM data for the data subcarriers in the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax FETCh:OFDM:EVM:PEAK:DECibel:DATA?

Arguments None

Returns <NRf> data in dB.

Examples FETCH:OFDM:EVM:PEAK:DECIBEL:DATA? might return –20.1872549032 indicating the data is –20.19 dB for the measurement.

FETCh:OFDM:EVM:PEAK:DECibel:PILots? (Query Only)

Returns the peak EVM data for the pilot subcarriers in the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax FETCh:OFDM:EVM:PEAK:DECibel:PILots?

Arguments None

Returns <NRf> data in dB.

Examples `FETCH:OFDM:EVM:PEAK:DECIBEL:PILOTS?` might return `–20.2872549032` indicating the data is –20.29 dB for the measurement.

FETCh:OFDM:EVM:PEAK:PERCent:ALL? (Query Only)

Returns the peak EVM data for all subcarriers in the OFDM measurement as a percent.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:EVM:PEAK:PERCent:ALL?`

Arguments None

Returns <NRf> data in percent.

Examples `FETCH:OFDM:EVM:PEAK:PERCENT:ALL?` might return `19.3223863840` indicating the data is 19.322% for the measurement.

FETCh:OFDM:EVM:PEAK:PERCent:DATA? (Query Only)

Returns the peak EVM data for the data subcarriers in the OFDM measurement as a percent.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:EVM:PEAK:PERCent:DATA?`

Arguments None

Returns <NRf> data in percent.

Examples FETCH:OFDM:EVM:PEAK:PERCENT:DATA? might return 19.3223863840 indicating the data is 19.322% for the measurement.

FETCh:OFDM:EVM:PEAK:PERCent:PILots? (Query Only)

Returns the peak EVM data for the pilot subcarriers in the OFDM measurement as a percent.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax FETCh:OFDM:EVM:PEAK:PERCent:PILots?

Arguments None

Returns <NRf> data in percent.

Examples FETCH:OFDM:EVM:PEAK:PERCENT:PILOTS? might return 10.8543863840 indicating the data is 10.854% for the measurement.

FETCh:OFDM:EVM:PEAK:SCARrier:ALL? (Query Only)

Returns the peak EVM data for all subcarriers at the subcarrier level in the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax FETCh:OFDM:EVM:PEAK:SCARrier:ALL?

Arguments None

Returns <NR1>

Examples `FETCH:OFDM:EVM:PEAK:SCARRIER:ALL?` might return `99.00000` indicating the data is 99 for the measurement.

FETCh:OFDM:EVM:PEAK:SCARrier:DATA? (Query Only)

Returns the peak EVM data for all data subcarriers at the subcarrier level in the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:EVM:PEAK:SCARrier:DATA?`

Arguments None

Returns <NR1>

Examples `FETCH:OFDM:EVM:PEAK:SCARRIER:DATA?` might return `99.00000` indicating the data is 99 for the measurement.

FETCh:OFDM:EVM:PEAK:SCARrier:PILots? (Query Only)

Returns the peak EVM data for the pilot subcarriers at the subxcarrier level in the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:EVM:PEAK:SCARrier:PILots?`

Arguments None

Returns <NR1>

Examples FETCH:OFDM:EVM:PEAK:SCARRIER:PILOTS? might return 88.00000 indicating the data is 88for the measurement

FETCh:OFDM:EVM:PEAK:SYMBol:ALL? (Query Only)

Returns the peak EVM data for all subcarriers at the symbol level in the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax FETCh:OFDM:EVM:PEAK:SYMBol:ALL?

Arguments None

Returns <NR1>

Examples FETCH:OFDM:EVM:PEAK:SYMBOL:ALL? might return 2.00000 indicating the data is 2 for the measurement

FETCh:OFDM:EVM:PEAK:SYMBol:DATA? (Query Only)

Returns the peak EVM data for the data subcarriers at the symbol level in the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax FETCh:OFDM:EVM:PEAK:SYMBol:DATA?

Arguments None

Returns <NR1>

Examples `FETCH:OFDM:EVM:PEAK:SYMBOL:DATA?` might return `2.00000` indicating the data is 2 for the measurement

FETCh:OFDM:EVM:PEAK:SYMBol:PILots? (Query Only)

Returns the peak EVM data for the pilot subcarriers at the symbol level in the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:EVM:PEAK:SYMBol:PILots?`

Arguments None

Returns <NR1>

Examples `FETCH:OFDM:EVM:PEAK:SYMBOL:PILOTS?` might return `1.00000` indicating the data is 1for the measurement

FETCh:OFDM:EVM:RMS:DECibel:ALL? (Query Only)

Returns the RMS EVM data for all subcarriers in the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:EVM:RMS:DECibel:ALL?`

Arguments None

Returns <NRf> data in dB.

Examples `FETCH:OFDM:EVM:RMS:DECIBEL:ALL?` might return `-26.9012093267` indicating the data is –26.90 for the measurement.

FETCh:OFDM:EVM:RMS:DECibel:DATA? (Query Only)

Returns the RMS EVM data for the data subcarriers in the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:EVM:RMS:DECibel:DATA?`

Arguments None

Returns <NRf> data in dB.

Examples `FETCH:OFDM:EVM:RMS:DECIBEL:DATA?` might return `-26.8477116269` indicating the data is –26.85 for the measurement.

FETCh:OFDM:EVM:RMS:DECibel:PILots? (Query Only)

Returns the RMS EVM data for the pilot subcarriers in the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:EVM:RMS:DECibel:PILots?`

Arguments None

Returns <NRf> data in dB.

Examples `FETCH:OFDM:EVM:RMS:DECIBEL:PILOTS?` might return `-33.0589143032` indicating the data is –33.06 dB.

FETCh:OFDM:EVM:RMS:PERCent:ALL? (Query Only)

Returns the peak RMS data for all subcarriers in the OFDM measurement as a percent.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:EVM:RMS:PERCent:ALL?`

Arguments None

Returns <NRf> data in percent.

Examples `FETCH:OFDM:EVM:RMS:PERCENT:ALL?` might return `225.0743627548` indicating the data is 225.074% for the measurement.

FETCh:OFDM:EVM:RMS:PERCent:DATA? (Query Only)

Returns the peak RMS data for the data subcarriers in the OFDM measurement as a percent.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:EVM:RMS:PERCent:DATA?`

Arguments None

Returns <NRf> data in percent.

Examples `FETCH:OFDM:EVM:RMS:PERCENT:DATA?` might return `260.6973409653` indicating that the data is 260.697% for the measurement.

FETCh:OFDM:EVM:RMS:PERCent:PILots? (Query Only)

Returns the peak RMS data for the pilot subcarriers in the OFDM measurement as a percent.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:EVM:RMS:PERCent:PILots?`

Arguments None

Returns <NRf> data in percent.

Examples `FETCH:OFDM:EVM:RMS:PERCENT:PILOTS?` might return `210.8103863840` indicating the data is 210.810% for the measurement.

FETCh:OFDM:EVM:TRACe<x>? (Query Only)

Returns the EVM trace data for the OFDM measurement.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbols.

When <x> is 3, the parameter is Average versus Subcarrier.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:EVM:TRACe<x>?`

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.
<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:OFDM:EVM:TRACE2?` might return `#252xxxx. . .` (52 byte data) for the Average versus Symbols measurement.

FETCh:OFDM:FERRor? (Query Only)

Returns the Frequency error reading for the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:FERRor?`

Arguments None

Returns <NRf> frequency error.

Examples `FETCH:OFDM:FERROR?` might return `82.8617142098E+3` indicating the frequency error was 82.86 kHz.

FETCh:OFDM:FLATness:PASS? (Query Only)

Queries whether the average power levels of subcarriers across the signal bandwidth on the OFDM Spectral Flatness display remain within the limits defined for a particular standard.

Conditions Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group Fetch commands

Syntax `FETCh:OFDM:FLATness:PASS?`

Returns Returns "1" in the case that all data remains within the defined limits, and "0" otherwise.

Examples `FETCH:OFDM:FLAT:PASS?` might return 1, which means that the average power levels of subcarriers across the signal bandwidth remain within the defined limits.

FETCh:OFDM:FLATness:RESult? (Query Only)

Returns the summary results of each segment of the OFDM Spectral Flatness display.

Conditions Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group Fetch commands

Syntax `FETCh:OFDM:FLATness:RESult?`

Returns The data is returned as 12 comma delineated text values. The specific values are :

- minimum subcarrier within segment 1
- minimum subcarrier average energy deviation from the global average within segment 1
- deviation the of minimum subcarrier average from the allowed value in segment 1
- maximum subcarrier within segment 1
- maximum subcarrier average energy deviation from the global average within segment 1
- deviation of the maximum subcarrier average from the allowed value in segment 1
- minimum subcarrier within segment 2
- minimum subcarrier average energy deviation from the global average within segment 2
- deviation the of minimum subcarrier average from the allowed value in segment 2
- maximum subcarrier within segment 2
- maximum subcarrier average energy deviation from the global average within segment 2
- deviation of the maximum subcarrier average from the allowed value in segment 2

Examples `FETCH:OFDM:FLAT:RES?` might return `-6.0000000000,-17.0626174659E-3,3.9829373825,12.0000000000,19.979260` which represents the summary results of each segment of the OFDM Spectral Flatness display.

FETCh:OFDM:FLATness:TRACe<x>? (Query Only)

Returns the OFDM Spectral Flatness trace data. When <x> is 1, the parameter is Matrix (symbol deviation per subcarrier). When<x> is 2, the parameter is Average Deviation vs Subcarrier.

Conditions Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group Fetch commands

Syntax FETCh:OFDM:FLATness:TRACe<x>?

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where:

<num_digit> is the number of digits in <num_byte.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples FETCH:OFDM:FLAT:TRACE2? might return #510400.. (10400-byte data) which represents the data for the Average Deviation vs. Subcarrier trace.

FETCh:OFDM:GAIN:IMBalance? (Query Only)

Returns the gain imbalance for the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax FETCh:OFDM:GAIN:IMBalance?

Arguments None

Returns <NRf> gain imbalance in dB.

Examples FETCH:OFDM:GAIN:IMBALANCE? might return -57.746E-3 indicating that the gain imbalance is –0.057746 dB.

FETCh:OFDM:IQ:ORIGin:OFFSet? (Query Only)

Returns the IQ origin offset for the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:IQ:ORIGin:OFFSet?`

Arguments None

Returns <NRf> gain imbalance in dB.

Examples `FETCH:OFDM:IQ:ORIGIN:OFFSET?` might return `–53.47017` indicating that the IQ origin offset is –53.47 dB.

FETCh:OFDM:MERRor:TRACe<x>? (Query Only)

Returns the magnitude error trace data for the OFDM measurement.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbols.

When <x> is 3, the parameter is Average versus Subcarrier.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:MERRor:TRACe<x>?`

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the data value at the n^{th} data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:OFDM:MERROR:TRACE2` might return `#3108xxxx. . .` (108-byte data) for the average versus symbols trace measurement.

FETCh:OFDM:PACKet:DIRection? (Query Only)

Returns the direction of the packet in the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:PACKet:DIRection?`

Arguments None

Returns One of the following values.

`UP` specifies that the packet is in the up direction.

`DOWN` specifies that the packet is in the down direction.

`BOTH` specifies that the packet is both directions.

Examples `FETCH:OFDM:PACKET:DIRECTION?` might return `DOWN` indicating that the packet was in the down direction.

FETCh:OFDM:PERRor:TRACe<x>? (Query Only)

Returns the phase error trace data for the OFDM measurement.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbols.

When <x> is 3, the parameter is Average versus Subcarrier.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:PERRor:TRACe<x>?`

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the data value at the n^{th} data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:OFDM:PERROR:TRACE2` might return `#3100xxxx. . .` (100-byte data) for the average versus symbols trace measurement.

FETCh:OFDM:POWer:TRACe<x>? (Query Only)

Returns the power trace data for the OFDM measurement.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbols.

When <x> is 3, the parameter is Average versus Subcarrier.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:POWer:TRACe<x>?`

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the data value at the n^{th} data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:OFDM:POWER:TRACE2` might return `#296xxxx. . .` (96-byte data) for the average versus symbols trace measurement.

FETCh:OFDM:QUADrature:OFFSet? (Query Only)

Returns the quadrature offset in the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:QUADrature:OFFSet?`

Arguments None

Returns <NRf> quadrature offset in degrees.

Examples `FETCH:OFDM:QUADRATURE:OFFSET?` might return `–99.9` indicating that the offset is –99°.

FETCh:OFDM:SCARriers? (Query Only)

Returns the number of subcarriers in the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:SCARriers?`

Arguments None

Returns <NR1> number of subcarriers.

Examples `FETCH:OFDM:SCARRIERS?` might return `200.0000` indicating that there are 200 subcarriers in the measurement.

FETCh:OFDM:SCARriers:SPACing? (Query Only)

Returns the subcarrier spacing for the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:SCARriers:SPACing?`

Arguments None

Returns <NRf> subcarrier spacing.

Examples `FETCH:OFDM:SCARRIERS:SPACING?` might return `90.000E+3` indicating that the subcarrier spacing is 90.000 kHz.

FETCh:OFDM:STABle:VALUe? (Query Only)

Returns the number of bytes in the symbol table for the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:STABle:VALUe?`

Arguments None

Returns #<num_digit><num_byte>

Where

<num_digit> is the number of digits in <num_byte>

<num_byte> is the number of bytes of data.

Examples `FETCH:OFDM:STABLE:VALUE?` might return `#43848` indicating that there are 3848 bytes in the symbol table.

FETCh:OFDM:SYMBol:CERRor? (Query Only)

Returns the symbol clock error for the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:SYMBol:CERRor?`

Arguments None

Returns <NRf> symbol clock error in ppm.

Examples `FETCH:OFDM:SYMBOL:CERROR?` might return `422.7135479929` indicating that the symbol clock error was 422.714 ppm.

FETCh:OFDM:SYMBol:COUNt? (Query Only)

Returns the number of symbols for the OFDM measurement.

Conditions Measurement views: OFDM

Group Fetch commands

Syntax `FETCh:OFDM:SYMBol:COUNt?`

Arguments None

Returns <NR1> number of symbols in the measurement.

Examples `FETCH:OFDM:SYMBOL:COUNT?` might return `25.00000` indicating that there were 25 symbols for the measurement.

FETCh:P25:CONSte:FERRor? (Query Only)

Returns the frequency error in Hz. The frequency error is the difference between the measured carrier frequency of the signal and the user-selected center frequency of the analyzer.

When Frequency Error is in Auto mode, this query returns the frequency error value. When in Manual mode, it returns the frequency offset.

Conditions Measurement view: P25 Constellation

Group Fetch commands

Syntax `FETCh:P25:CONSte:FERRor?`

Returns <freq_error> :: <NRf>, which is the frequency error in Hz.

Examples `FETCH:P25:CONSTE:FERROR?` might return -10.7E+3, which is a frequency error of -10.7 kHz.

FETCh:P25:CONSte:TRACe? (Query Only)

Returns the P25 Constellation display trace data.

Conditions Measurement view: P25 Constellation

Group Fetch commands

Syntax `FETCh:P25:CONSte:TRACe?`

Returns #<num_digit><num_byte><I(1)><Q(1)><I(2)><Q(2)>...<I(n)><Q(n)>

Where:

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<I(n)> and <Q(n)> are the normalized I- and Q- coordinate values at the *n*th data point. 4-byte little endian floating point format specified in IEEE 488.2.

Examples `FETCH:P25:CONSTE:TRACE?` might return #43848xxxx... (3848-byte data) for the P25 Constellation trace data.

FETCh:P25:EDIagram:FDEViation? (Query Only)

Returns the Frequency Deviation vs Time trace data with the X value.

Conditions Measurement view: P25 Eye Diagram

Group Fetch commands

Syntax `FETCh:P25:EDIagram:FDEViation?`

Returns #<num_digit><num_byte><Y(1)><X(1)><Y(2)><X(2)>...<Y(n)><X(n)>

Where:

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<Y(n)> and <X(n)> are the frequency deviation in Hz and time (symbols) coordinate pair at the *n*th data point. 4-byte little endian floating point format specified in IEEE 488.2.

Examples `FETCH:P25:EDIAGRAM:FDEVIATION?` might return #3160xxxx... (160-byte data) for the Frequency Deviation vs Time trace.

FETCh:P25:EDIagram:FERRor? (Query Only)

Returns the frequency error in the P25 Eye Diagram display.

When Frequency Error is in Auto mode, this query returns the frequency error value. When in Manual mode, it returns the frequency offset.

Conditions Measurement view: P25 Eye Diagram

Group Fetch commands

Syntax `FETCh:P25:EDIagram:FERRor?`

Returns <feq_error>::<NRf> is the frequency error in Hz.

Examples `FETCH:P25:EDIAGRAM:FERROR?` might return -10.7E+3, indicating the frequency Error is -10.7 kHz.

FETCh:P25:EDIagram:I? (Query Only)

Returns the I versus Time trace data in the P25 Eye Diagram display.

Conditions Measurement view: P25 Eye Diagram

Group Fetch commands

Syntax `FETCh:P25:EDIagram:I?`

Returns #<num_digit><num_byte><Y(1)><X(1)><Y(2)><X(2)>...<Y(n)><X(n)>

Where:

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<Y(n)> and <X(n)> are the normalized I- and time (symbols) coordinate values at the *n*th data point. 4-byte little endian floating point format specified in IEEE 488.2.

Examples `FETCH:P25:EDIAGRAM:I?` might return #3160xxxx... (160-byte data) for the I versus Time trace.

FETCh:P25:EDIagram:Q? (Query Only)

Returns the Q versus Time trace data in the P25 Eye Diagram display.

Conditions Measurement view: P25 Eye Diagram

Group Fetch commands

Syntax `FETCh:P25:EDIagram:Q?`

Returns #<num_digit><num_byte><Y(1)><X(1)><Y(2)><X(2)>...<Y(n)><X(n)>

Where:

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<Y(n)> and <X(n)> are the normalized Q- and time (symbols) coordinate values at the *n*th data point. 4-byte little endian floating point format specified in IEEE 488.2.

Examples `FETCH:P25:EDIAGRAM:Q?` might return #3160xxxx... (160-byte data) for the Q Versus Time trace.

FETCh:P25:PVTime:BURSt:WIDTh? (Query Only)

Returns the value of the measured time width of the burst packet from poweron to powerdown (or the end of waveform, if that occurs before powerdown), in seconds, for the P25 Power vs. Time trace.

Conditions Measurement view: P25 Power vs. Time

Group Fetch commands

Syntax `FETCh:P25:PVTime:BURSt:WIDTh?`

Returns Floating point number that represents the value of the measured time width of the burst packet from power on to power down (or the end of waveform, if that occurs before power down), in seconds.

Examples `FETCH:P25:PVTIME:BURST:WIDTH?` might return 156.0000000000E-6, which represents a width of the burst packet of 156 µs.

FETCh:P25:PVTime:FALL:TIME:STARt? (Query Only)

Returns the value of the start time of the falling edge, in seconds.

Conditions Measurement view: P25 Power vs. Time

Group Fetch commands

Syntax `FETCh:P25:PVTime:FALL:TIME:STARt?`

Related Commands FETCh:P25:PVTime:FALL:TIME:STOP?

Returns Floating point number which represents the value of the start time of the falling edge of the burst, in seconds.

Examples `FETCH:P25:PVTIME:FALL:TIME:START?` might return 6e-6, which indicates that the start time of the falling edge is at 6 µs.

FETCh:P25:PVTime:FALL:TIME:STOP? (Query Only)

Returns the value of the stop time of the falling edge, in seconds.

Conditions Measurement view: P25 Power vs. Time

Group Fetch commands

Syntax `FETCh:P25:PVTime:FALL:TIME:STOP?`

Related Commands FETCh:P25:PVTime:FALL:TIME:STARt?

Returns Floating point number which represents the value of the stop time of the falling edge of the burst, in seconds.

Examples `FETCH:P25:PVTIME:FALL:TIME:STOP?` might return 7e-6, which indicates the stop time of the falling edge is at 7 µs.

FETCh:P25:PVTime:RISE:TIME? (Query Only)

Returns the value of the width of the rising edge, in seconds.

Conditions Measurement view: P25 Power vs. Time

Group Fetch commands

Syntax `FETCh:P25:PVTime:RISE:TIME?`

Related Commands FETCh:P25:PVTime:RTIMe?

Returns Floating point number which represents the value of the start time of the rising edge of the burst, in seconds.

Examples `FETCH:P25:PVTIME:RISE:TIME?` might return 6e-6, which indicates the start time of the rising edge is 6 µs.

FETCh:P25:PVTime:RISE:TIME:STARt? (Query Only)

Returns the value of the start time of the rising edge, in seconds.

Conditions Measurement view: P25 Power vs. Time

Group Fetch commands

Syntax `FETCh:P25:PVTime:RISE:TIME:STARt?`

Related Commands FETCh:P25:PVTime:RISE:TIME?

FETCh:P25:PVTime:RTIMe?

Returns Floating point number which represents the value of the start time of the rising edge of the burst, in seconds.

Examples `FETCH:P25:PVTIME:RISE:TIME:START?` might return 6e-6, which indicates the start time of the rising edge is at 6 µs.

FETCh:P25:PVTime:RISE:TIME:STOP? (Query Only)

Returns the value of the stop time of the rising edge, in seconds.

Conditions Measurement view: P25 Power vs. Time

Group Fetch commands

Syntax `FETCh:P25:PVTime:RISE:TIME:STOP?`

Related Commands FETCh:P25:PVTime:RISE:TIME:STARt?

FETCh:P25:PVTime:RTIMe?

FETCh:P25:PVTime:RISE:TIME?

Returns Floating point number which represents the value of the stop time of the rising edge of the burst, in seconds.

Examples `FETCH:P25:PVTIME:RISE:TIME:STOP?` might return 7e-6, which indicates the width of the rising edge is 7 µs.

FETCh:P25:PVTime:RTIMe? (Query Only)

Returns the value of the width of the rising edge, in seconds.

Conditions Measurement view: P25 Power vs. Time

Group Fetch commands

Syntax `FETCh:P25:PVTime:RTIMe?`

Related Commands FETCh:P25:PVTime:RISE:TIME?

Returns Floating point number which represents the value of the width of the rising edge of the burst, in seconds.

Examples `FETCH:P25:PVTIME:RTIME?` might return 7e-6, which indicates the width of the rising edge is 7 µs.

FETCh:P25:PVTime:TRACe[:Y]? (Query Only)

Returns the vertical values (power) for the P25 Power vs. Time trace.

Conditions Measurement view: P25 Power vs. Time

Group Fetch commands

Syntax `FETCh:P25:PVTime:TRACe[:Y]?`

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where:

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the *n*th data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:P25:PVTIME:TRACE[:Y]?` might return #43204 (3204-byte data), which represents the vertical values (power) for the P25 Power vs. Time trace.

FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF? (Query Only)

Returns the HCPM Pmax-off measurement results.

Conditions Measurement view: P25 Summary

Group Fetch commands

Syntax `FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF?`

Returns HCPM Pmax-off measurement results.

Examples `FETCH:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MAXOFF?` would return the HCPM Pmax-off measurement result.

FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON? (Query Only)

Returns the HCPM Pmax-on measurement results.

Conditions Measurement view: P25 Summary

Group Fetch commands

Syntax `FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON?`

Returns HCPM Pmax-on measurement results.

Examples `FETCH:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MAXON?` would return HCPM Pmax-on measurement results.

FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSS? (Query Only)

Returns the HCPM Pss-max measurement results.

Conditions Measurement view: P25 Summary

Group Fetch commands

Syntax `FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSS?`

Returns HCPM Pss-max measurement results.

Examples `FETCH:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MAXSS?` would return the HCPM Pss-max measurement result.

FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum? (Query Only)

Returns the HCPM Pss-min measurement results.

Conditions Measurement view: P25 Summary

Group Fetch commands

Syntax `FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum?`

Returns HCPM Pss-min measurement results.

Examples `FETCH:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MINIMUM?` would return the HCPM Pss-min measurement result.

FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:OFFSlot? (Query Only)

Returns the HCPM Offslot power results.

Conditions Measurement view: P25 Summary

Group Fetch commands

Syntax `FETCH:P25:SUMMary:POWEr:MEASurement:HCPM:OFFSlot?`

Returns HCPM Offslot power results.

Examples `FETCH:P25:SUMMARY:POWER:MEASUREMENT:HCPM:OFFSLOT?` would return the HCPM Offslot power result.

FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE? (Query Only)

Returns the HCPM Time alignment's t_error_1 measurement results.

Conditions Measurement view: P25 Summary

Group Fetch commands

Syntax `FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE?`

Returns HCPM Time alignment's t_error_1 measurement results.

Examples `FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TERROR:ONE?`

FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE:AC (Query Only)

Returns the HCPM Time alignment's t_error_1 acquisition count over which the t_error_1 average is computed.

Conditions Measurement view: P25 Summary

Group Fetch commands

Syntax
```
FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:
TERRor:ONE:ACQ:COUNT?
```

Returns HCPM Time alignment's t_error_1 acquisition count.

Examples
```
FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:
TERROR:ONE:ACQ:COUNT?
```

FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO? (Query Only)

Returns the HCPM Time alignment's t_error_0 measurement results.

Conditions Measurement view: P25 Summary

Group Fetch commands

Syntax
```
FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:
TERRor:ZERO?
```

Returns HCPM Time alignment's t_error_0 measurement results.

Examples
```
FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:
TERROR:ZERO?
```

FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO:ACQ (Query Only)

Returns the HCPM Time alignment's t_error_0 acquisition count over which the t_error_0 average is done.

Conditions Measurement view: P25 Summary

Group Fetch commands

Syntax
```
FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:
TERRor:ZERO:ACQ:COUNT?
```

Returns HCPM Time alignment's t_error_0 acquisition count.

Examples
```
FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:
TERROR:ZERO:ACQ:COUNT?
```

FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TOBSync? (Query Only)

Returns the HCPM Time alignment's t_obsync measurement results.

Conditions Measurement view: P25 Summary

Group Fetch commands

Syntax
```
FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:
TOBSync?
```

Returns HCPM Time alignment's t_obsync measurement results.

Examples
```
FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:
TOBSYNC?
```

FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TOBSync:ACQ:C (Query Only)

Returns the HCPM Time alignment's t_obsync acquisition counts over which the t_obsync average is computed.

Conditions Measurement view: P25 Summary

Group Fetch commands

Syntax
```
FETCH:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:
TOBSync:ACQ:COUNT?
```

Returns HCPM Time alignment's t_obsync acquisition counts.

Examples
```
FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:
TOBSYNC:ACQ:COUNT?
```

FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT? (Query Only)

Returns the Average Transmitter Power Attack acquisition count over which the t_obsync average is computed.

Conditions Measurement view: P25 Summary

Group Fetch commands

Syntax
```
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:
ACQ:COUNT?
```

Returns Average Transmitter Power Attack acquisition count.

Examples
```
FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:
ACQ:COUNT?
```

FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT:BI? (Query Only)

Returns the Average Transmitter Power Attack, busy/idle, acquisition count over which the Busy/Idle Attack time average is computed.

Conditions Measurement view: P25 Summary

Group Fetch commands

Syntax

```
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:
ACQ:COUNT:BI?
```

Returns Average Transmitter Power Attack, busy/idle, acquisition count over which the Busy/Idle Attack time average is computed.

Examples

```
FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:
ACQ:COUNT:BI?
```

FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder? (Query Only)

Returns the Average Transmitter Encoder Attack Time measurement results.

Conditions Measurement view: P25 Summary

Group Fetch commands

Syntax

```
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:
ENCOder?
```

Returns Average Transmitter Encoder Attack Time measurement results.

Examples

```
FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:
ENCODER?
```

FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder:BI? (Query Only)

Returns the Average Transmitter Encoder Attack Time, busy/idle, measurement results.

Conditions Measurement view: P25 Summary

Group Fetch commands

Syntax
```
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder:BI?
```

Returns Average Transmitter Encoder Attack Time , busy/idle, measurement results.

Examples
```
FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:ENCODER:BI?
```

FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr? (Query Only)

Returns the Average Transmitter Power Attack Time measurement results.

Conditions Measurement view: P25 Summary

Group Fetch commands

Syntax
```
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr?
```

Returns Average Transmitter Power Attack Time measurement results.

Examples
```
FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:POWER?
```

FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI? (Query Only)

Returns the Average Transmitter Power Attack Time, busy/idle, measurement results.

Conditions Measurement view: P25 Summary

Group Fetch commands

Syntax

```
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:
POWEr:BI?
```

Returns Average Transmitter Power Attack Time, busy/idle, measurement results.

Examples

```
FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:
POWER:BI?
```

FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy? (Query Only)

Returns the Throughput delay measurement results.

Conditions Measurement view: P25 Summary

Group Fetch commands

Syntax

```
FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy?
```

Returns Throughput delay measurement results.

Examples

```
FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:THRUPUT:DELAY?
```

FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy:ACQ:COUNT? (Query Only)

Returns the Throughput delay acquisition counts over which the Throughput delay average is computed.

Conditions Measurement view: P25 Summary

Group Fetch commands

Syntax `FETCH:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy:ACQ:COUNT?`

Returns Throughput delay acquisition counts.

Examples `FETCH:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:THRUPUT:DELAY:ACQ:COUNT?`

FETCh:PERRor:FERRor? (Query Only)

Returns the frequency error in the Phase error versus Time measurement.

Conditions Measurement views: Phase error versus Time

Group Fetch commands

Syntax `FETCh:PERRor:FERRor?`

Arguments None

Returns `<freq_error> ::` `<NRf>` is the frequency error in Hz.

Examples `FETCH:PERROR:FERROR?` might return `-10.7E+3`, indicating the frequency error is -10.7 kHz.

FETCh:PERRor:PEAK? (Query Only)

Returns the peak value in the Phase error versus Time measurement.

Conditions Measurement views: Phase error versus Time

Group Fetch commands

Syntax `FETCh:PERRor:PEAK?`

Related Commands FETCh:PERRor:PINDex?

Arguments None

Returns `<peak> ::` `<NRf>` is the peak phase error in degrees.

Examples `FETCH:PERROR:PEAK?` might return `0.683`, indicating the peak phase error is 0.683 °.

FETCh:PERRor:PINDex? (Query Only)

Returns the time at the phase error peak.

Conditions Measurement views: Phase error versus Time

Group Fetch commands

Syntax `FETCh:PERRor:PINDex?`

Related Commands FETCh:PERRor:PEAK?, [SENSe]:DDEMod:TIMe:UNITs

Arguments None

Returns `<peak_time> ::` `<NRf>` is the time at the phase error peak in symbol number. The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples `FETCH:PERROR:PINDEX?` might return `68.000`, indicating that the phase error peak is at symbol #68.

FETCh:PERRor:RMS? (Query Only)

Returns the RMS (Root-Mean-Square) value in the Phase error versus Time measurement.

Conditions Measurement views: Phase error versus Time

Group Fetch commands

Syntax `FETCh:PERRor:RMS?`

Arguments None

Returns `<rms> ::  <NRf>` is the RMS phase error in degrees.

Examples `FETCH:PERROR:RMS?` might return `0.746`, indicating the phase error is 0.746 ° RMS.

FETCh:PERRor:TRACe? (Query Only)

Returns the Phase error versus Time trace data.

Conditions Measurement views: Phase error versus Time

Group Fetch commands

Syntax `FETCh:PERRor:TRACe?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.
<data(n)> is the phase error in degrees at the nth data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples FETCH:PERROR:TRACE? might return #42036xxxx... (2036-byte data) for the Phase error versus Time trace.

FETCh:PHVTime? (Query Only)

Returns the Phase versus Time trace data.

Conditions Measurement views: Phase versus Time

Group Fetch commands

Syntax FETCh:PHVTime?

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the phase in degrees at the nth data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples FETCH:PHVTIME? might return #3160xxxx... (160-byte data) for the Phase versus Time trace.

FETCh:PHVTime:MAXimum? (Query Only)

Returns the maximum value in the Phase versus Time measurement.

Conditions Measurement views: Phase versus Time

Group Fetch commands

Syntax `FETCh:PHVTime:MAXimum?`

Related Commands FETCh:PHVTime:MAXLocation?

Arguments None

Returns `<max> ::` `<NRf>` is the maximum phase in degrees.

Examples `FETCH:PHVTIME:MAXIMUM?` might return `153.8`, indicating the maximum phase is 153.8 °.

FETCh:PHVTime:MAXLocation? (Query Only)

Returns the time at which the phase is maximum.

Conditions Measurement views: Phase versus Time

Group Fetch commands

Syntax `FETCh:PHVTime:MAXLocation?`

Related Commands FETCh:PHVTime:MAXimum?

Arguments None

Returns `<max_time> ::` `<NRf>` is the time in seconds at which the phase is maximum.

Examples `FETCH:PHVTIME:MAXLOCATION?` might return `175.3E-9`, indicating the I or Q level is maximum at 175.3 ns.

FETCh:PHVTime:MINimum? (Query Only)

Returns the minimum value in the Phase versus Time measurement.

Conditions Measurement views: Phase versus Time

Group Fetch commands

Syntax `FETCh:PHVTime:MINimum?`

Related Commands FETCh:PHVTime:MINLocation?

Arguments None

Returns `<min> ::` `<NRf>` is the minimum phase in degrees.

Examples `FETCH:PHVTIME:MINIMUM?` might return `-176.3`, indicating the minimum phase is -176.3 °.

FETCh:PHVTime:MINLocation? (Query Only)

Returns the time at which the phase is minimum.

Conditions Measurement views: Phase versus Time

Group Fetch commands

Syntax `FETCh:PHVTime:MINLocation?`

Related Commands FETCh:PHVTime:MINimum?

Arguments None

Returns `<min_time> ::` `<NRf>` is the time in seconds at which the phase is minimum.

Examples `FETCH:PHVTIME:MINLOCATION?` might return `450.7E-9`, indicating the phase is minimum at 450.7 ns.

FETCh:PHVTime:RESult? (Query Only)

Returns the Phase versus Time measurement results.

Conditions Measurement views: Phase versus Time

Group Fetch commands

Syntax `FETCh:PHVTime:RESult?`

Arguments None

Returns `<max>,<max_time>,<min>,<min_time>`

Where
`<max> ::` `<NRf>` is the maximum phase in degrees.
`<max_time> ::` `<NRf>` is the time in seconds at which the phase is maximum.
`<min> ::` `<NRf>` is the minimum phase in degrees.
`<min_time> ::` `<NRf>` is the time in seconds at which the phase is minimum.

Examples `FETCH:PHVTIME:RESULT?` might return `153.8,175.3E-9,-176.3,450.7E-9`, indicating the maximum phase is 153.8 ° at 175.3 ns and the minimum phase is -176.3 ° at 450.7 ns.

FETCh:PM:PNEGative? (Query Only)

Returns the negative peak phase deviation (–Pk) in the PM measurement.

Conditions Measurement views: PM

Group Fetch commands

Syntax `FETCh:PM:PNEGative?`

Arguments None

Returns `<-Pk> ::` `<NRf>` is the negative peak phase deviation in degrees.

Examples `FETCH:PM:PNEGATIVE?` might return `–23.42`, indicating the positive peak phase deviation is –23.42 °.

FETCh:PM:PPOSitive? (Query Only)

Returns the positive peak phase deviation (+Pk) in the PM measurement.

Conditions Measurement views: PM

Group Fetch commands

Syntax `FETCh:PM:PPOSitive?`

Arguments None

Returns `<+Pk> ::` `<NRf>` is the positive peak phase deviation in degrees.

Examples `FETCH:PM:PPOSITIVE?` might return `26.87`, indicating the positive peak phase deviation is 26.87 °.

FETCh:PM:PTPeak? (Query Only)

Returns the peak-peak phase deviation (Pk-Pk) in the PM measurement.

Conditions Measurement views: PM

Group Fetch commands

Syntax `FETCh:PM:PTPeak?`

Arguments None

Returns `<Pk-Pk> ::` `<NRf>` is the peak-peak phase deviation in degrees.

Examples `FETCH:PM:PTPEAK?` might return `46.34`, indicating the peak-peak phase deviation is 46.34 °.

FETCh:PM:RESult? (Query Only)

Returns the PM measurement results.

Conditions Measurement views: PM

Group Fetch commands

Syntax `FETCh:PM:RESult?`

Arguments None

Returns `<+Pk>,<–Pk>,<RMS>,<Pk-Pk>`

Where
`<+Pk> ::` `<NRf>` is the positive peak phase deviation in degrees.
`<-Pk> ::` `<NRf>` is the negative peak phase deviation in degrees.
`<RMS> ::` `<NRf>` is the RMS phase deviation in degrees.
`<Pk-Pk> ::` `<NRf>` is the peak-peak phase deviation in degrees.

Examples `FETCH:PM:RESULT?` might return `22.89,–23.45,15.12,46.34`.

FETCh:PM:RMS? (Query Only)

Returns the RMS phase deviation in the PM measurement.

Conditions Measurement views: PM

Group Fetch commands

Syntax `FETCh:PM:RMS?`

Arguments None

Returns `<RMS> ::` `<NRf>` is the RMS phase deviation in degrees.

Examples `FETCH:PM:RMS?` might return `15.12`, indicating the RMS frequency deviation is 15.12 °.

FETCh:PNOise:ALL? (Query Only)

Returns all results of the phase noise measurement.

Conditions Measurement views: Phase noise

Group Fetch commands

Syntax `FETCh:PNOise:ALL?`

Arguments `None`

Returns `<Cpower>,<Ferror>,<Pnoise>,<Tjitter>,<Rjitter>,<RFM>`

Where
`<Cpower> ::` `<NRf>` is the carrier power in dBm.
`<Ferror> ::` `<NRf>` is the frequency error in Hz.
`<Pnoise> ::` `<NRf>` is the RMS phase noise in degrees.
`<Tjitter> ::` `<NRf>` is the total jitter in seconds.
`<Rjitter> ::` `<NRf>` is the random jitter in seconds.
`<RFM> ::` `<NRf>` is the residual FM in Hz.

Examples `FETCH:PNOISE:ALL?` might return `-9.455,1.235E+6,51.43,2.312E-9,4.178E-9,14.58`, indicating
Carrier power: -9.455 dBm,
Frequency error: 1.235 MHz,
RMS phase noise: 51.43 °,
Total jitter: 2.312 ns,
Random jitter: 4.178 ns, and
Residual FM: 14.58 Hz.

FETCh:PNOise:CARRier:FERRor? (Query Only)

Returns the carrier frequency error in the phase noise measurement.

Conditions Measurement views: Phase noise

Group Fetch commands

Syntax `FETCh:PNOise:CARRier:FERRor?`

Arguments None

Returns `<NRf>` Carrier frequency error in Hz.

Examples `FETCH:PNOISE:CARRIER:FERROR?` might return `1.235E+6`, indicating that the carrier frequency error is 1.235 MHz.

FETCh:PNOise:CARRier:POWer? (Query Only)

Returns the carrier power in the phase noise measurement.

Conditions Measurement views: Phase noise

Group Fetch commands

Syntax `FETCh:PNOise:CARRier:POWer?`

Arguments None

Returns `<NRf>` Carrier power in dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `FETCH:PNOISE:CARRIER:POWER?` might return `-9.455`, indicating that the carrier power is -9.455 dBm.

FETCh:PNOise:JITTer? (Query Only)

Returns the jitter in the phase noise measurement.

Conditions Measurement views: Phase noise

Group Fetch commands

Syntax `FETCh:PNOise:JITTer?`

Arguments None

Returns <NRf> Jitter in seconds.

Examples `FETCH:PNOISE:JITTER?` might return `2.312E-9`, indicating that the jitter is 2.312 ns.

FETCh:PNOise:RESidual:FM? (Query Only)

Returns the residual FM in the phase noise measurement.

Conditions Measurement views: Phase noise

Group Fetch commands

Syntax `FETCh:PNOise:RESidual:FM?`

Arguments None

Returns <NRf> Residual FM in Hz.

Examples `FETCH:PNOISE:RESIDUAL:FM?` might return `14.58`, indicating that the residual FM is 14.58 Hz.

FETCh:PNOise:RMS:PNOise? (Query Only)

Returns the RMS phase noise in the phase noise measurement.

Conditions Measurement views: Phase noise

Group Fetch commands

Syntax `FETCh:PNOise:RMS:PNOise?`

Arguments None

Returns <NRf> RMS phase noise in degrees.

Examples FETCH:PNOISE:RMS:PNOISE? might return 51.43, indicating that the RMS phase noise is 51.43 °.

FETCh:PNOise:SPECtrum<x>:X? (Query Only)

Returns the frequencies of the specified trace.

The parameter <x> = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions Measurement views: Phase noise

Group Fetch commands

Syntax FETCh:PNOise:SPECtrum<x>:X?

Arguments None

Returns #<num_digit><num_byte><x(1)><x(2)>...<x(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<x(n)> is the frequency (Hz) at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples FETCH:PNOISE:SPECTRUM1:X? might return #516020xxxx... (16020-byte data) for the frequencies of Trace 1.

FETCh:PNOise:SPECtrum<x>:XY? (Query Only)

Returns the frequency and phase noise pairs of the specified trace.

The parameter <x> = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions Measurement views: Phase noise

Group Fetch commands

Syntax `FETCh:PNOise:SPECtrum<x>:XY?`

Arguments None

Returns `#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>`

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<x(n)><y(n)> is the frequency (Hz) and phase noise (dBc/Hz) pair at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:PNOISE:SPECTRUM1:XY?` might return `#516020xxxx...` (16020-byte data) for the frequency and phase noise pairs of the Trace 1.

FETCh:PNOise:SPECtrum<x>[:Y]? (Query Only)

Returns the phase noise values of the specified trace.

The parameter <x> = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions Measurement views: Phase noise

Group Fetch commands

Syntax `FETCh:PNOise:SPECtrum<x>[:Y]?`

Arguments None

Returns `#<num_digit><num_byte><y(1)><y(2)>...<y(n)>`

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<y(n)> is the phase noise (dBc/Hz) at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:PNOISE:SPECTRUM1:Y` might return `#516020xxxx...` (16020-byte data) for the phase noise values of Trace 1.

FETCh:PULSe[:RESult]:ATX? (Query Only)

Returns the average transmitted power in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax `FETCh:PULSe[:RESult]:ATX?`

Arguments None

Returns `<first_pulse_num>,<ATX(1)>,< ATX(2)>,...<ATX(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<ATX(n)> ::` `<NRf>` is the average transmitted power for the pulse with the number of [first_pulse_num + n - 1] in dBm.
The unit can be changed to watts by the [SENSe]:POWer:UNITs command.
Volt is invalid in the average transmitted power measurement.

Examples `FETCH:PULSE:RESULT:ATX?` might return `0,-18.57,-18.73,-18.20,-18.53` for Pulse 0 to 3.

FETCh:PULSe[:RESult]:AVERage? (Query Only)

Returns the average on power in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax `FETCh:PULSe[:RESult]:AVERage?`

Arguments None

Returns

<first_pulse_num>,< avg(1)>,< avg(2)>,...<avg(n)>

Where
<first_pulse_num> :: <NR1> is the first pulse number.
<avg(n)> :: <NRf> is the average on power for the pulse with the number of [first_pulse_num + n - 1] in dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples

FETCH:PULSE:RESULT:AVERAGE? might return 0,-2.354,-2.368,-2.343,-2.358 for Pulse 0 to 3.

FETCh:PULSe[:RESult]:DRODb? (Query Only)

Returns the droop in dB in the pulse table.

Conditions

Measurement views: Pulse table

Group

Fetch commands

Syntax

FETCh:PULSe[:RESult]:DRODb?

Related Commands

FETCh:PULSe[:RESult]:DROop?

Arguments

None

Returns

<first_pulse_num>,<droop(1)>,<droop(2)>,...<droop(n)>

Where
<first_pulse_num> :: <NR1> is the first pulse number.
<droop(n)> :: <NRf> is the wattage droop for the pulse with the number of [first_pulse_num + n - 1] in dB.

Examples

FETCH:PULSE:RESULT:DRODB? might return 1,-0.4,0.32,0.2,-0.1 for Pulse 1 to 4.

FETCh:PULSe[:RESult]:DROop? (Query Only)

Returns the droop in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax `FETCh:PULSe[:RESult]:DROop?`

Related Commands FETCh:PULSe[:RESult]:DRODb?

Arguments None

Returns `<first_pulse_num>,<droop(1)>,<droop(2)>,...<droop(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<droop(n)> ::` `<NRf>` is the wattage droop for the pulse with the number of [first_pulse_num + n - 1] in percent (%).

Examples `FETCH:PULSE:RESULT:DROOP?` might return `0,-270.9E-3,-193.0E-3, -242.7E-3,-177.5E-3` for Pulse 0 to 3.

FETCh:PULSe[:RESult]:DUTPct? (Query Only)

Returns the duty factor (%) in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax `FETCh:PULSe[:RESult]:DUTPct?`

Arguments None

Returns `<first_pulse_num>,<duty_pct(1)>,<duty_pct(2)>,... <duty_pct(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<duty_pct(n)> ::` `<NRf>` is the duty factor for the pulse with the number of [first_pulse_num + n - 1] in percent (%).

Examples FETCH:PULSE:RESULT:DUTPCT? might return 0,28.94,28.96,29.00,29.01 for Pulse 0 to 3.

FETCh:PULSe[:RESult]:DUTRatio? (Query Only)

Returns the duty factor (ratio) in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax FETCh:PULSe[:RESult]:DUTRatio?

Arguments None

Returns <first_pulse_num>,<duty_ratio(1)>,<duty_ratio(2)>,...<duty_ratio(n)>

Where
<first_pulse_num> :: <NR1> is the first pulse number.
<duty_ratio(n)> :: <NRf> is the duty factor for the pulse with the number of [first_pulse_num + n - 1] (no unit).

Examples FETCH:PULSE:RESULT:DUTRATIO? might return 0,289.4E-3,289.6E-3,290.0E-3,290.1E-3 for Pulse 0 to 3.

FETCh:PULSe[:RESult]:FALL? (Query Only)

Returns the fall time in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax FETCh:PULSe[:RESult]:FALL?

Arguments None

Returns `<first_pulse_num>,<fall(1)>,<fall(2)>,...<fall(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<fall(n)> ::` `<NRf>` is the fall time for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples `FETCH:PULSE:RESULT:FALL?` might return `0,110.3E-9,90.45E-9, 95.03E-9,111.9E-9` for Pulse 0 to 3.

FETCh:PULSe[:RESult]:FDELta? (Query Only)

Returns the computed delta frequency from the carrier in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax `FETCh:PULSe[:RESult]:FDELta?`

Arguments None

Returns `<first_pulse_num>,<freq_delta(1)>,<freq_delta(2)>,... <freq_delta(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<freq_delta(n)> ::` `<NRf>` is the frequency delta for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples `FETCH:PULSE:RESULT:FDELTA?` might return `1.0000000000,163.6692962646,82.8298492432,21.8995475769,89.296356201` for the first pulse and each pulse found.

FETCh:PULSe[:RESult]:FRDeviation? (Query Only)

Returns the frequency deviation in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax `FETCh:PULSe[:RESult]:FRDeviation?`

Arguments None

Returns `<first_pulse_num>,<freq_dev(1)>,<freq_dev(2)>,...<freq_dev(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<freq_dev(n)> ::` `<NRf>` is the frequency deviation for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples `FETCH:PULSE:RESULT:FRDEVIATION?` might return `1,740.6E+3, 736.5E+3,718.3E+3,672.2E+3` for Pulse 1 to 4.

FETCh:PULSe[:RESult]:IRAMplitude? (Query Only)

Returns the pulse impulse response amplitude measurement in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax `FETCh:PULSe[:RESult]:IRAMplitude?`

Arguments None

Returns `<first_pulse_num>,<amplitude(1)>,<amplitude(2)>,...<amplitude(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<amplitude(n)> ::` `<NRf>` is the sidelobe level, in dB, from the mainlobe for the pulse with the number of [first_pulse_num + n - 1].

Examples `FETCH:PULSE:RESULT:IRAMPLITUDE?` might return `1.0000000000,-45.3,-47.5,-50.2` for Pulse 1 through 3.

FETCh:PULSe[:RESult]:IRTime? (Query Only)

Returns the impulse response time in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax `FETCh:PULSe[:RESult]:IRTime?`

Returns `<first_pulse_num>, <time(1>), <time(2)>,....<time(n)>`

Where

`<first_pulse_num> ::` `<NR1>` is the first pulse number.

`<time(n)> ::` `<NRf>` is the sidelobe offset time from the mainlobe for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples `FETCH:PULSE:RESULT:IRTIME?` might return `1.0000000000,806.6708687693E-9,-846.6595318168E-9,-873.3477443457E-9` for Pulse 1 through 3.

FETCh:PULSe[:RESult]:MFReqerror? (Query Only)

Returns the maximum frequency error in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax `FETCh:PULSe[:RESult]:MFReqerror?`

Arguments None

Returns `<first_pulse_num>,<max_freq_err(1)>,<max_freq_err(2)>,...<max_freq_err(n)>`

Where

`<first_pulse_num> ::` `<NR1>` is the first pulse number.

<max_freq_err(n)> :: <NRf> is the maximum frequency error for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples FETCH:PULSE:RESULT:MFREQERROR? might return 1,597.5E+3,675.8E+3,642.8E+3,598.2E+3 for Pulse 1 to 4.

FETCh:PULSe[:RESult]:MPHerror? (Query Only)

Returns the maximum phase error in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax FETCh:PULSe[:RESult]:MPHerror?

Arguments None

Returns <first_pulse_num>,<max_phase_err(1)>,<max_phase_err(2)>,... <max_phase_err(n)>

Where
<first_pulse_num> :: <NR1> is the first pulse number.
<max_phase_err(n)> :: <NRf> is the maximum phase error for the pulse with the number of [first_pulse_num + n - 1] in degrees.

Examples FETCH:PULSE:RESULT:MPHERROR? might return 1,-9.221,-8.413, -11.853,-10.258 for Pulse 1 to 4.

FETCh:PULSe[:RESult]:OVEDb? (Query Only)

Returns the overshoot in dB in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax FETCh:PULSe[:RESult]:OVEDb?

Related Commands FETCh:PULSe[:RESult]:OVERshoot?

Arguments None

Returns `<first_pulse_num>,<overshoot(1)>,<overshoot(2)>,...<overshoot(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<overshoot(n)> ::` `<NRf>` is the overshoot for the pulse with the number of [first_pulse_num + n - 1] in dB.

Examples `FETCH:PULSE:RESULT:OVERDB?` might return `1,1.2,2.3,1.0,0.2` for Pulse 1 to 4.

FETCh:PULSe[:RESult]:OVERshoot? (Query Only)

Returns the overshoot in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax `FETCh:PULSe[:RESult]:OVERshoot?`

Related Commands FETCh:PULSe[:RESult]:OVEDb?

Arguments None

Returns `<first_pulse_num>,<overshoot(1)>,<overshoot(2)>,...<overshoot(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<overshoot(n)> ::` `<NRf>` is the overshoot for the pulse with the number of [first_pulse_num + n - 1] in percent (%) of watts or volts.

Examples `FETCH:PULSE:RESULT:OVERSHOOT?` might return `1,1.2,2.3,1.0,0.2` for Pulse 1 to 4.

FETCh:PULSe[:RESult]:PHDeviation? (Query Only)

Returns the phase deviation in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax `FETCh:PULSe[:RESult]:PHDeviation?`

Arguments None

Returns `<first_pulse_num>,<phase_dev(1)>,<phase_dev(2)>,...<phase_dev(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<phase_dev(n)> ::` `<NRf>` is the phase deviation for the pulse with the number of [first_pulse_num + n - 1] in degrees.

Examples `FETCH:PULSE:RESULT:PHDEVIATION?` might return `1,11.658,9.640,10.509,8.272` for Pulse 1 to 4.

FETCh:PULSe[:RESult]:PPFRequency? (Query Only)

Returns the pulse-pulse carrier frequency in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax `FETCh:PULSe[:RESult]:PPFRequency?`

Arguments None

Returns `<first_pulse_num>,<pp_freq(1)>,<pp_freq(2)>,...<pp_freq(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.

<pp_freq(n)> :: <NRf> is the pulse-pulse carrier frequency for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples FETCH:PULSE:RESULT:PPFREQUENCY? might return 0,0.000,1.258E+3,-3.121E+3,1.862E+3 for Pulse 0 to 3.

FETCh:PULSe[:RESult]:PPOWer? (Query Only)

Returns the peak power in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax FETCh:PULSe[:RESult]:PPOWer?

Arguments None

Returns <first_pulse_num>,<pk_power(1)>,<pk_power(2)>,...<pk_power(n)>

Where
<first_pulse_num> :: <NR1> is the first pulse number.
<pk_power(n)> :: <NRf> is the peak power for the pulse with the number of [first_pulse_num + n - 1] in dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples FETCH:PULSE:RESULT:PPOWER? might return 0,-2.26,-2.27,-2.23,-2.25 for Pulse 0 to 3.

FETCh:PULSe[:RESult]:PPPHase? (Query Only)

Returns the pulse-pulse carrier phase in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax `FETCh:PULSe[:RESult]:PPPHase?`

Arguments None

Returns `<first_pulse_num>,<pp_phase(1)>,<pp_phase(2)>,...<pp_phase(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<pp_phase(n)> ::` `<NRf>` is the pulse-pulse carrier phase for the pulse with the number of [first_pulse_num + n - 1] in degrees.

Examples `FETCH:PULSE:RESULT:PPPHASE?` might return `0,0.000,21.66,46.76,57.56` for Pulse 0 to 3.

FETCh:PULSe[:RESult]:RINTerval? (Query Only)

Returns the repetition interval in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax `FETCh:PULSe[:RESult]:RINTerval?`

Arguments None

Returns `<first_pulse_num>,<rep_int(1)>,<rep_int(2)>,...<rep_int(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<rep_int(n)> ::` `<NRf>` is the repetition interval for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples `FETCH:PULSE:RESULT:RINTERVAL?` might return `0,16.03E-6,16.08E-6,16.07E-6,16.02E-6` for Pulse 0 to 3.

FETCh:PULSe[:RESult]:RIPDb? (Query Only)

Returns the ripple in dB in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax `FETCh:PULSe[:RESult]:RIPDb?`

Related Commands FETCh:PULSe[:RESult]:RIPPle?

Arguments None

Returns `<first_pulse_num>,<ripple(1)>,<ripple(2)>,...<ripple(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<ripple(n)> ::` `<NRf>` is the ripple for the pulse with the number of [first_pulse_num + n - 1] in dB.

Examples `FETCH:PULSE:RESULT:RIPDB?` might return `1,0.4,0.32,0.2,0.1` for Pulse 1 to 4.

FETCh:PULSe[:RESult]:RIPPle? (Query Only)

Returns the ripple in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax `FETCh:PULSe[:RESult]:RIPPle?`

Related Commands FETCh:PULSe[:RESult]:RIPDb?

Arguments None

Returns `<first_pulse_num>,<ripple(1)>,<ripple(2)>,...<ripple(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<ripple(n)> ::` `<NRf>` is the volts or watts ripple for the pulse with the number of [first_pulse_num + n - 1] in percent (%).

Examples `FETCH:PULSE:RESULT:RIPPLE?` might return `0,106.5E-3,177.6E-3, 148.3E-3,148.5E-3` for Pulse 0 to 3.

FETCh:PULSe[:RESult]:RISE? (Query Only)

Returns the rise time in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax `FETCh:PULSe[:RESult]:RISE?`

Arguments None

Returns `<first_pulse_num>,<rise(1)>,<rise(2)>,...<rise(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<rise(n)> ::` `<NRf>` is the rise time for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples `FETCH:PULSE:RESULT:RISE?` might return `0,92.94E-9,115.9E-9, 115.1E-9,97.45E-9` for Pulse 0 to 3.

FETCh:PULSe[:RESult]:RMSFreqerror? (Query Only)

Returns the RMS frequency error in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax `FETCh:PULSe[:RESult]:RMSFreqerror?`

Arguments None

Returns `<first_pulse_num>,<RMS_freq_err(1)>,<RMS_freq_err(2)>,...<RMS_freq_err(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<RMS_freq_err(n)> ::` `<NRf>` is the RMS frequency error for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples `FETCH:PULSE:RESULT:RMSFREQERROR?` might return `1,51.54E+3,69.20E+3,64.21E+3,51.02E+3` for Pulse 1 to 4.

FETCh:PULSe[:RESult]:RMSPherror? (Query Only)

Returns the RMS phase error in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax `FETCh:PULSe[:RESult]:RMSPherror?`

Arguments None

Returns `<first_pulse_num>,<RMS_phase_err(1)>,<RMS_phase_err(2)>,...<RMS_phase_err(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<RMS_phase_err(n)> ::` `<NRf>` is the RMS phase error for the pulse with the number of [first_pulse_num + n - 1] in degrees.

Examples `FETCH:PULSE:RESULT:RMSPHERROR?` might return `1,908.4E-3,752.8E-3,981.7E-3,886.4E-3` for Pulse 1 to 4.

FETCh:PULSe[:RESult]:RRATe? (Query Only)

Returns the repetition rate in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax `FETCh:PULSe[:RESult]:RRATe?`

Arguments None

Returns `<first_pulse_num>,<rep_rate(1)>,<rep_rate(2)>,...<rep_rate(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<rep_rate(n)> ::` `<NRf>` is the repetition rate for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples `FETCH:PULSE:RESULT:RRATE?` might return `0,62.50E+3,62.52E+3,62.51E+3,62.49E+3` for Pulse 0 to 3.

FETCh:PULSe[:RESult]:TIME? (Query Only)

Returns the time in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax `FETCh:PULSe[:RESult]:TIME?`

Arguments None

Returns `<first_pulse_num>,<time(1)>,<time(2)>,...<time(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.

<time(n)> :: <NRf> is the time for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples FETCH:PULSE:RESULT:TIME? might return 1,7.937E-3,8.436E-3, 6.504E-3,9.876E-3 for Pulse 1 to 4.

FETCh:PULSe[:RESult]:WIDTh? (Query Only)

Returns the pulse width in the pulse table.

Conditions Measurement views: Pulse table

Group Fetch commands

Syntax FETCh:PULSe[:RESult]:WIDTh?

Arguments None

Returns <first_pulse_num>,<width(1)>,<width(2)>,...<width(n)>

Where
<first_pulse_num> :: <NR1> is the first pulse number.
<width(n)> :: <NRf> is the pulse width for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples FETCH:PULSE:RESULT:WIDTH? might return 0,4.630E-6,4.632E-6, 4.639E-6,4.642E-6 for Pulse 0 to 3.

FETCh:PULSe:STATistics:ATX? (Query Only)

Returns the average transmitted power in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax FETCh:PULSe:STATistics:ATX?

Arguments None

Returns <ATX_avg>,<ATX_min>,<ATX_max>

Where
<ATX_avg> :: <NRf> is the average of the average transmitted power.
<ATX_min> :: <NRf> is the minimum of the average transmitted power.
<ATX_max> :: <NRf> is the maximum of the average transmitted power.
Unit: dBm.
The unit can be changed to watts by the [SENSe]:POWer:UNITs command.
Volt is invalid in the average transmitted power measurement.

Examples FETCH:PULSE:STATISTICS:ATX? might return -18.51,-18.74,-18.12 for the average transmitted power in the pulse statistics.

FETCh:PULSe:STATistics:AVERage? (Query Only)

Returns the average on power in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax FETCh:PULSe:STATistics:AVERage?

Arguments None

Returns <avg_avg>,<avg_min>,<avg_max>

Where
<avg_avg> :: <NRf> is the average of the average on power.
<avg_min> :: <NRf> is the minimum of the average on power.
<avg_max> :: <NRf> is the maximum of the average on power.
Unit: dBm.
The unit can be changed to watts by the [SENSe]:POWer:UNITs command.

Examples FETCH:PULSE:STATISTICS:AVERAGE? might return -2.35,-2.36,-2.34 for the average on power in the pulse statistics.

FETCh:PULSe:STATistics:DRODb? (Query Only)

Returns the droop in dB in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax `FETCh:PULSe:STATistics:DRODb?`

Related Commands FETCh:PULSe:STATistics:DROop?

Arguments None

Returns `<droop_avg>,<droop_min>,<droop_max>`

Where
`<droop_avg>` :: `<NRf>` is the average droop.
`<droop_min>` :: `<NRf>` is the minimum droop.
`<droop_max>` :: `<NRf>` is the maximum droop.
Unit: in dB.

Examples `FETCH:PULSE:STATISTICS:DRODB?` might return `22.67E-3,-613.5E-3,633.8E-3` for the droop in the pulse statistics.

FETCh:PULSe:STATistics:DROop? (Query Only)

Returns the droop in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax `FETCh:PULSe:STATistics:DROop?`

Related Commands FETCh:PULSe:STATistics:DRODb?

Arguments None

Returns `<droop_avg>,<droop_min>,<droop_max>`

Where
`<droop_avg> ::` `<NRf>` is the average droop.
`<droop_min> ::` `<NRf>` is the minimum droop.
`<droop_max> ::` `<NRf>` is the maximum droop.
Unit: Percent (%) by watts.

Examples `FETCH:PULSE:STATISTICS:DROOP?` might return `22.67E-3,-613.5E-3,633.8E-3` for the droop in the pulse statistics.

FETCh:PULSe:STATistics:DUTPct? (Query Only)

Returns the duty factor (%) in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax `FETCh:PULSe:STATistics:DUTPct?`

Arguments None

Returns `<duty_pct_avg>,<duty_pct_min)>,<duty_pct_max>`

Where
`<duty_pct_avg> ::` `<NRf>` is the average duty factor.
`<duty_pct_min> ::` `<NRf>` is the minimum duty factor.
`<duty_pct_max> ::` `<NRf>` is the maximum duty factor.
Unit: Percent (%).

Examples `FETCH:PULSE:STATISTICS:DUTPCT?` might return `2.437,2.310,2.657` for the duty factor (%) in the pulse statistics.

FETCh:PULSe:STATistics:DUTRatio? (Query Only)

Returns the duty factor (ratio) in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax `FETCh:PULSe:STATistics:DUTRatio?`

Arguments None

Returns `<duty_ratio_avg>,<duty_ratio_min>,<duty_ratio_max)>`

Where
`<duty_ratio_avg>` :: `<NRf>` is the average duty factor.
`<duty_ratio_min>` :: `<NRf>` is the minimum duty factor.
`<duty_ratio_max>` :: `<NRf>` is the maximum duty factor.
Unit: None.

Examples `FETCH:PULSE:STATISTICS:DUTRATIO?` might return `24.37E-3,23.11E-3, 26.57E-3` for the duty factor (ratio) in the pulse statistics.

FETCh:PULSe:STATistics:FALL? (Query Only)

Returns the fall time in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax `FETCh:PULSe:STATistics:FALL?`

Arguments None

Returns `<fall_avg>,<fall_min>,<fall_max>`

Where
<fall_avg> :: <NRf> is the average fall time.
<fall_min> :: <NRf> is the minimum fall time.
<fall_max> :: <NRf> is the maximum fall time.
Unit: Seconds.

Examples FETCH:PULSE:STATISTICS:FALL? might return 70.27E-9,69.62E-9, 71.27E-9 for the fall time in the pulse statistics.

FETCh:PULSe:STATistics:FDELta? (Query Only)

Return the delta frequency in the pulse statistics.

Group Fetch commands

Syntax FETCh:PULSe:STATistics:FDELta?

Arguments None

Returns <freq_delta_avg>,<freq_delta_min>,<freq_delta_max>

Where
<freq_delta_avg> :: <NRf> is the average frequency deviation.
<freq_delta_min> :: <NRf> is the minimum frequency deviation.
<freq_delta_max> :: <NRf> is the maximum frequency deviation.
Unit: Hz.

Examples FETCH:PULSE:STATISTICS:FDELTA? might return 77.9785690308,-110.1940994263,287.8957519531 for the frequency delta in the pulse statistics.

FETCh:PULSe:STATistics:FRDeviation? (Query Only)

Returns the frequency deviation in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax `FETCh:PULSe:STATistics:FRDeviation?`

Arguments None

Returns `<freq_dev_avg>,<freq_dev_min>,<freq_dev_max>`

Where
`<freq_dev_avg> ::` `<NRf>` is the average frequency deviation.
`<freq_dev_min> ::` `<NRf>` is the minimum frequency deviation.
`<freq_dev_max> ::` `<NRf>` is the maximum frequency deviation.
Unit: Hz.

Examples `FETCH:PULSE:STATISTICS:FRDEVIATION?` might return `754.1E+3,660.5E+3,835.2E+3` for the frequency deviation in the pulse statistics.

FETCh:PULSe:STATistics:IRAMplitude? (Query Only)

Returns the Pulse Impulse Response Amplitude measurement in the pulse statistics.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax `FETCh:PULSe:STATistics:IRAMplitude?`

Arguments None

Returns `<amp_avg>,<amp_min>,<amp_max>`

Where

`<amp_avg> ::` `<NRf>` is the average of the impulse response amplitude.

`<amp_min> ::` `<NRf>` is the minimum of the impulse response amplitude.

`<amp_max> ::` `<NRf>` is the maximum of impulse response amplitude.

Unit: dB.

Examples `FETCH:PULSE:STATISTICS:IRAMPLITUDE?` might return `-45.3,-47.1,-43.2` for the impulse response amplitude in the pulse statistics.

FETCh:PULSe:STATistics:IRTime? (Query Only)

Returns the impulse response time in the pulse statistics.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax `FETCh:PULSe:STATistics:IRTime?`

Returns `<avg_avg>,<avg_min>,<avg_max>`

Where

`<avg_avg> ::` `<NRf>` is the average of the impulse response time.

`<avg_min> ::` `<NRf>` is the minimum of the impulse response time.

`<avg_max> ::` `<NRf>` is the maximum of impulse response time.

Unit: seconds.

Examples `FETCH:PULSE:STATISTICS:IRTIME?` might return `-2.35,-2.36,-2.34` for the impulse response time in the pulse statistics.

FETCh:PULSe:STATistics:MFReqerror? (Query Only)

Returns the maximum frequency error in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax `FETCh:PULSe:STATistics:MFReqerror?`

Arguments None

Returns <max_freq_err_avg>,<max_freq_err_min>,<max_freq_err_max>

Where
<max_freq_err_avg> :: <NRf> is the average of maximum frequency error.
<max_freq_err_min> :: <NRf> is the minimum of maximum frequency error.
<max_freq_err_max> :: <NRf> is the maximum of maximum frequency error.
Unit: Hz.

Examples FETCH:PULSE:STATISTICS:MFREQERROR? might return 645.0E+3,555.6E+3,738.8E+3 for the maximum frequency error in the pulse statistics.

FETCh:PULSe:STATistics:MPHerror? (Query Only)

Returns the maximum phase error in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax FETCh:PULSe:STATistics:MPHerror?

Arguments None

Returns <max_phase_err_avg>,<max_phase_err_min>,<max_phase_err_max>

Where
<max_phase_err_avg> :: <NRf> is the average of maximum phase error.
<max_phase_err_min> :: <NRf> is the minimum of maximum phase error.
<max_phase_err_max> :: <NRf> is the maximum of maximum phase error.
Unit: Degrees.

Examples FETCH:PULSE:STATISTICS:MPHERROR? might return -11.47,-17.18,-7.61 for the maximum phase error in the pulse statistics.

FETCh:PULSe:STATistics:OVEDb? (Query Only)

Returns the overshoot measurement result in dB in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd or TTRend.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax `FETCh:PULSe:STATistics:OVEDb?`

Related Commands FETCh:PULSe:STATistics:OVERshoot?

Arguments None

Returns `<overshoot_avg>,<overshoot_min>,<overshoot_max>`

Where
`<overshoot_avg>` :: `<NRf>` is the average overshoot.
`<overshoot_min>` :: `<NRf>` is the minimum overshoot.
`<overshoot_max>` :: `<NRf>` is the maximum overshoot.
Unit: dB.

Examples `FETCH:PULSE:STATISTICS:OVEDB?` might return `0.3,0.1,0.5` for the overshoot measurement result in dB in the pulse statistics.

FETCh:PULSe:STATistics:OVERshoot? (Query Only)

Returns the overshoot in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd or TTRend.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax `FETCh:PULSe:STATistics:OVERshoot?`

Related Commands FETCh:PULSe:STATistics:OVEDb?

Arguments None

Returns <overshoot_avg>,<overshoot_min>,<overshoot_max>

Where
<overshoot_avg> :: <NRf> is the average overshoot.
<overshoot_min> :: <NRf> is the minimum overshoot.
<overshoot_max> :: <NRf> is the maximum overshoot.
Unit: Percent (%) by watts or volts.

Examples FETCH:PULSE:STATISTICS:OVERSHOOT? might return 0.3,0.1,0.5 for the overshoot in the pulse statistics.

FETCh:PULSe:STATistics:PHDeviation? (Query Only)

Returns the phase deviation in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax FETCh:PULSe:STATistics:PHDeviation?

Arguments None

Returns <phase_dev_avg>,<phase_dev_min>,<phase_dev_max>

Where
<phase_dev_avg> :: <NRf> is the average phase deviation.
<phase_dev_min> :: <NRf> is the minimum phase deviation.
<phase_dev_max> :: <NRf> is the maximum phase deviation.
Unit: Degrees.

Examples FETCH:PULSE:STATISTICS:PHDEVIATION? might return 11.678,7.694, 17.374 for the phase deviation in the pulse statistics.

FETCh:PULSe:STATistics:PPFRequency? (Query Only)

Returns the pulse-pulse carrier frequency in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax `FETCh:PULSe:STATistics:PPFRequency?`

Arguments None

Returns `<pp_freq_avg>,<pp_freq_min)>,<pp_freq_max>`

Where
`<pp_freq_avg> ::` `<NRf>` is the average pulse-pulse carrier frequency.
`<pp_freq_min> ::` `<NRf>` is the minimum pulse-pulse carrier frequency.
`<pp_freq_max> ::` `<NRf>` is the maximum pulse-pulse carrier frequency.
Unit: Hz.

Examples `FETCH:PULSE:STATISTICS:PPFREQUENCY?` might return `1.135E+3,311.3E+3,-262.8E+3` for the pulse-pulse carrier frequency in the pulse statistics.

FETCh:PULSe:STATistics:PPOWer? (Query Only)

Returns the peak power in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax `FETCh:PULSe:STATistics:PPOWer?`

Arguments None

Returns `<pk_power_avg>,<pk_power_min>,<pk_power_max>`

Where
`<pk_power_avg> ::` `<NRf>` is the average peak power.
`<pk_power_min> ::` `<NRf>` is the minimum peak power.
`<pk_power_max> ::` `<NRf>` is the maximum peak power.
Unit: dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `FETCH:PULSE:STATISTICS:PPOWER?` might return `-2.273,-2.313,-2.235` for the peak power in the pulse statistics.

FETCh:PULSe:STATistics:PPPHase? (Query Only)

Returns the pulse-pulse carrier phase in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax `FETCh:PULSe:STATistics:PPPHase?`

Arguments None

Returns `<pp_phase_avg>,<pp_phase_min)>,<pp_phase_max>`

Where
`<pp_phase_avg> ::` `<NRf>` is the average pulse-pulse carrier phase.
`<pp_phase_min> ::` `<NRf>` is the minimum pulse-pulse carrier phase.
`<pp_phase_max> ::` `<NRf>` is the maximum pulse-pulse carrier phase.
Unit: Degrees.

Examples `FETCH:PULSE:STATISTICS:PPPHASE?` might return `-9.298E-3,-254.3E-3,311.7E-3` for the pulse-pulse carrier phase in the pulse statistics.

FETCh:PULSe:STATistics:RINTerval? (Query Only)

Returns the repetition interval in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax `FETCh:PULSe:STATistics:RINTerval?`

Arguments None

Returns `<rep_int_avg>,<rep_int_min>,<rep_int_max>`

Where
`<rep_int_avg> ::` `<NRf>` is the average repetition interval.
`<rep_int_min> ::` `<NRf>` is the minimum repetition interval.
`<rep_int_max> ::` `<NRf>` is the maximum repetition interval.
Unit: Seconds.

Examples `FETCH:PULSE:STATISTICS:RINTERVAL?` might return `240.5E-6,217.9E-6,281.2E-6` for the repetition interval in the pulse statistics.

FETCh:PULSe:STATistics:RIPDb? (Query Only)

Returns the ripple in dB in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax `FETCh:PULSe:STATistics:RIPDb?`

Related Commands FETCh:PULSe:STATistics:RIPPle?

Arguments None

Returns `<ripple_avg>,<ripple_min>,<ripple_max>`

Where

<ripple_avg> :: <NRf> is the average ripple.
<ripple_min> :: <NRf> is the minimum ripple.
<ripple_max> :: <NRf> is the maximum ripple.
Unit: dB.

Examples FETCH:PULSE:STATISTICS:RIPDB? might return 160.4E-3,83.78E-3,287.7E-3 for the ripple in dB in the pulse statistics.

FETCh:PULSe:STATistics:RIPPle? (Query Only)

Returns the ripple in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax FETCh:PULSe:STATistics:RIPPle?

Related Commands FETCh:PULSe:STATistics:RIPDb?

Arguments None

Returns <ripple_avg>,<ripple_min>,<ripple_max>

Where
<ripple_avg> :: <NRf> is the average ripple.
<ripple_min> :: <NRf> is the minimum ripple.
<ripple_max> :: <NRf> is the maximum ripple.
Unit: Percent (%) by watts or volts.

Examples FETCH:PULSE:STATISTICS:RIPPLE? might return 160.4E-3,83.78E-3,287.7E-3 for the ripple in the pulse statistics.

FETCh:PULSe:STATistics:RISE? (Query Only)

Returns the rise time in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax `FETCh:PULSe:STATistics:RISE?`

Arguments None

Returns `<rise_avg>,<rise_min>,<rise_max>`

Where
`<rise_avg> ::` `<NRf>` is the average rise time.
`<rise_min> ::` `<NRf>` is the minimum rise time.
`<rise_max> ::` `<NRf>` is the maximum rise time.
Unit: Seconds.

Examples `FETCH:PULSE:STATISTICS:RISE?` might return `105.4E-9,91.65E-9, 116.2E-9` for the rise time in the pulse statistics.

FETCh:PULSe:STATistics:RMSFreqerror? (Query Only)

Returns the RMS frequency error in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax `FETCh:PULSe:STATistics:RMSFreqerror?`

Arguments None

Returns `<RMS_freq_err_avg>,<RMS_freq_err_min>,<RMS_freq_err_max>`

Where
`<RMS_freq_err_avg> ::` `<NRf>` is the average of RMS frequency error.
`<RMS_freq_err_min> ::` `<NRf>` is the minimum of RMS frequency error.
`<RMS_freq_err_max> ::` `<NRf>` is the maximum of RMS frequency error.
Unit: Hz.

Examples `FETCH:PULSE:STATISTICS:RMSFREQERROR?` might return `63.67E+3,45.49E+3,81.28E+3` for the RMS frequency error in the pulse statistics.

FETCh:PULSe:STATistics:RMSPherror? (Query Only)

Returns the RMS phase error in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax `FETCh:PULSe:STATistics:RMSPherror?`

Arguments None

Returns `<RMS_phase_err_avg>,<RMS_phase_err_min>,<RMS_phase_err_max>`

Where
`<RMS_phase_err_avg>` :: `<NRf>` is the average of RMS phase error.
`<RMS_phase_err_min>` :: `<NRf>` is the minimum of RMS phase error.
`<RMS_phase_err_max>` :: `<NRf>` is the maximum of RMS phase error.
Unit: Degrees.

Examples `FETCH:PULSE:STATISTICS:RMSPHERROR?` might return `1.032,604.5E-3,1.606` for the RMS phase error in the pulse statistics.

FETCh:PULSe:STATistics:RRATe? (Query Only)

Returns the repetition rate in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse trace

Group Fetch commands

Syntax `FETCh:PULSe:STATistics:RRATe?`

Arguments None

Returns `<rep_rate_avg>,<rep_rate_min>,<rep_rate_max>`

Where
`<rep_rate_avg>` :: `<NRf>` is the average repetition rate.
`<rep_rate_min>` :: `<NRf>` is the minimum repetition rate.
`<rep_rate_max>` :: `<NRf>` is the maximum repetition rate.
Unit: Hz.

Examples `FETCH:PULSE:STATISTICS:RRATE?` might return `62.50E+3,62.49E+3, 62.52E+3` for the repetition rate in the pulse statistics.

FETCh:PULSe:STATistics:WIDTh? (Query Only)

Returns the pulse width in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse trace

Group Fetch commands

Syntax `FETCh:PULSe:STATistics:WIDTh?`

Arguments None

Returns `<width_avg>,<width_min>,<width_max>`

Where
`<width_avg>` :: `<NRf>` is the average pulse width.
`<width_min>` :: `<NRf>` is the minimum pulse width.
`<width_max>` :: `<NRf>` is the maximum pulse width.
Unit: Seconds.

Examples `FETCH:PULSE:STATISTICS:WIDTH?` might return `4.636E-6,4.630E-6, 4.643E-6` for the pulse width in the pulse statistics.

FETCh:PULSe:STATistics:X? (Query Only)

Returns the horizontal values of the trace data of the pulse statistics measurement selected by the DISPlay:PULSe:SELect:RESult command.

NOTE. *Select the plot type (Trend, FFT, Time Trend or Histogram) using the DISPlay:PULSe:STATistics:PLOT command before executing this query.*

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax `FETCh:PULSe:STATistics:X?`

Arguments None

Returns `#<num_digit><num_byte><data1><data2>....data<n>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the horizontal value of the statistics graph at the n^{th} data point.

The unit is trace number (Plot = Trend), Hz (Plot = FFT), second (Plot = Time Trend) or current measurement unit (Plot = Histogram).
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:PULSE:STATISTICS:X?` might return `#264xxxx... (64-byte data)` for the statistics trace of the pulse width measurement when DISPlay:PULSe:SELect: RESult is set to WIDTh.

FETCh:PULSe:STATistics:XY? (Query Only)

Returns the horizontal and vertical values of the trace data of the pulse statistics measurement selected by the DISPlay:PULSe:SELect:RESult command.

NOTE. *Select the plot type (Trend, FFT, Time Trend or Histogram) using the DISPlay:PULSe:STATistics:PLOT command before executing this query.*

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax `FETCh:PULSe:STATistics:XY?`

Arguments None

Returns `#<num_digit><num_byte><data1><data2>....data<n>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the horizontal value of the statistics graph at the n^{th} data point.
`<data(n+1)>` is the vertical value of the statistics graph at the n^{th} data point.

For horizontal values, the unit is trace number (Plot = Trend), Hz (Plot = FFT), second (Plot = Time Trend) or current measurement unit (Plot = Histogram).

For vertical values, the unit is current measurement unit (Plot = Trend or Time Trend), dB (Plot = FFT), count (Plot = Histogram)
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:PULSE:STATISTICS:XY?` might return `#264xxxx...` (64-byte data) for the statistics trace of the pulse width measurement when DISPlay:PULSe:SELect: RESult is set to WIDTh.

FETCh:PULSe:STATistics[:Y]? (Query Only)

Returns the trace data of the pulse statistics measurement selected by the DISPlay:PULSe:SELect:RESult command.

NOTE. *Select the plot type (Trend, FFT, Time Trend or Histogram) using the DISPlay:PULSe:STATistics:PLOT command before executing this query.*

Conditions Measurement views: Pulse statistics

Group Fetch commands

Syntax `FETCh:PULSe:STATistics[:Y]?`

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the amplitude at the nth data point.
The unit is current measurement unit (Plot = Trend or Time Trend), dB (Plot = FFT), count or percent (Plot = Histogram)
4-byte little endian floating-point format specified in IEEE 488.2.

The unit of power is selected by the [SENSe]:POWer:UNITs command.

Examples FETCH:PULSE:STATISTICS[:Y]? might return #264xxxx... (64-byte data) for the
statistics trace of the pulse width measurement when DISPlay:PULSe:SELect:RESult is set to WIDTh.

FETCh:PULSe:TRACe:X? (Query Only)

Returns the time values of the pulse trace. Use the DISPlay:PULSe:SELect:NUMBer command to select the pulse, and the DISPlay:PULSe:SELect:RESult command to select the measurement result.

Conditions Measurement views: Pulse trace

Group Fetch commands

Syntax FETCh:PULSe:TRACe:X?

Arguments None

Returns #<num_digit><num_byte><X(1)><X(2)>...<X(n)>

Where
<num_digit>is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<X(n)> is the time in seconds at the nth data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples FETCH:PULSE:TRACE:X? might return #43204xxxx... (3204-byte data) for the time values of the trace.

FETCh:PULSe:TRACe:XY? (Query Only)

Returns the horizontal (time) and vertical value pairs of the pulse trace. Use the DISPlay:PULSe:SELect:NUMBer command to select the pulse, and the DISPlay:PULSe:SELect:RESult command to select the measurement result.

Conditions Measurement views: Pulse trace

Group Fetch commands

Syntax `FETCh:PULSe:TRACe:XY?`

Arguments None

Returns `#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<x(n)><y(n)>` is the horizontal value (time in seconds) and vertical value pair at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

The vertical unit depends on measurement results: Hz for frequency error and deviation, degrees for phase error and deviation, otherwise dBm. The vertical unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `FETCH:PULSE:TRACE:XY?` might return `#43204xxxx...` (3204-byte data) for the horizontal (time) and vertical value pairs of the pulse trace.

FETCh:PULSe:TRACe[:Y]? (Query Only)

Returns the vertical values of the pulse trace. Use the DISPlay:PULSe:SELect:NUMBer command to select the pulse, and the DISPlay:PULSe:SELect:RESult command to select the measurement result.

Conditions Measurement views: Pulse trace

Group Fetch commands

Syntax `FETCh:PULSe:TRACe[:Y]?`

Arguments None

Returns `#<num_digit><num_byte><y(1)><y(2)>...<y(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<y(n)>` is the amplitude (dBm) at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.

The unit depends on measurement results: Hz for frequency error and deviation, degrees for phase error and deviation, otherwise dBm. The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `FETCH:PULSE:TRACE:Y?` might return `#43204xxxx...` (3204-byte data) for the vertical values of the pulse trace.

FETCh:RFIN:IQ? (Query Only)

Returns time-domain IQ data for a specific acquisition data record. You can set a range of IQ pairs optionally. The instrument needs to be in stopped mode. If not in stopped mode, it returns the execution error (-200).

Conditions Measurement views: All

Group Fetch commands

Syntax `FETCh:RFIN:IQ? <rec_ID>[,<begin_num>,<end_num>]`

Arguments `<rec_ID>` specifies the acquisition data record ID number.
`<begin_num>` specifies the beginning number of IQ pairs.
`<end_num>` specifies the end number of IQ pairs.

Use the FETCh:RFIN:RECord:IDS? query to get the beginning and end ID's of acquisition data records.

Returns `#<num_digit><num_byte><I(1)><Q(1)><I(2)><Q(2)>...<I(n)><Q(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.
<I(n)> and <Q(n)> are the time-domain IQ data pair.
4-byte little endian floating-point format specified in IEEE 488.2.

To find out the range of IQ pairs, use the FETCh:RFIN:IQ:HEADer? query. The returned value of <num_sample> is the number of IQ pairs.

NOTE. *Do not fetch a large number of IQ pairs because of memory limitation.*

Examples

FETCH:RFIN:IQ? 10,25,350 might return #43848xxxx... (3848-byte data) for the record #10, ranging from 25th to 350th IQ pair.

FETCh:RFIN:IQ:HEADer? (Query Only)

Returns the header information for a specific acquisition data record. The instrument needs to be in stopped mode. If not in stopped mode, it returns the execution error (-200).

Conditions

Measurement views: All

Group

Fetch commands

Syntax

FETCh:RFIN:IQ:HEADer? <rec_ID>

Arguments

<rec_ID> specifies the acquisition data record ID number.

Use the FETCh:RFIN:RECord:IDS? query to get the beginning and end ID's of acquisition data records.

Returns

,<sampling_freq>,<num_sample>,<center_freq>,<acq_BW>,
<ref_level>,<trig_pos>,<time_stamp>,<FastFrame_ID>

Where
 :: <NR3> is the span in Hz.
<sampling_freq> :: <NR3> is the sampling frequency in Hz.
<num_sample> :: <NR3> is the number of samples.
<center_freq> :: <NR3> is the center frequency in Hz.
<acq_BW> :: <NR3> is the acquisition bandwidth in Hz.
<ref_level> :: <NR3> is the reference level in dBm.
<trig_pos> :: <NR3> is the trigger position in seconds.
<time_stamp> :: <string> is the time stamp.
<FastFrame_ID> :: <NR1> is the Fast Frame ID: Zero or positive number.

If the Fast Frame is disabled, the ID will be negative.

Examples `FETCH:RFIN:IQ:HEADER? 10` might return `40.0E+6,50.0E+6,4.027E+3,1.5E+9,40.0E+6,0.0,20.242E-6,"10/31/2007 1118:32 AM",-1`, indicating
Span: 40 MHz,
Sampling frequency: 50 MHz,
Number of samples: 4027,
Center frequency: 1.5 GHz,
Acquisition bandwidth: 40 MHz,
Reference level: 0 dBm,
Trigger position: 20.242 µs,
Time stamp: 10/31/2007 1118:32 AM, and
Fast Frame ID: -1

FETCh:RFIN:IQ:SCALe? (Query Only)

Returns the internal RF linear data scaling factor contained in the .tiq file header. The scaling factor can be used to convert digital IQ output (Option 05) values into real IQ values.

Conditions Measurement views: All

Group Fetch commands

Syntax `FETCh:RFIN:IQ:SCALe?`

Related Commands FETCh:RFIN:RECord:IDS?

Arguments None

Returns `<NR3>` The RF linear data scaling factor.

Examples `FETCH:RFIN:IQ:SCALE?` might return `19.553E-6` for the scaling factor.

FETCh:RFIN:IQ:VHEader? (Query Only)

Returns the verbose header information for a specific acquisition data record. The instrument needs to be in stopped mode. If not in stopped mode, it returns the execution error (-200).

Conditions Measurement views: All

Group Fetch commands

Syntax `FETCh:RFIN:IQ:VHEader? <rec_ID>`

Arguments `<rec_ID>` specifies the acquisition data record ID number.

Use the FETCh:RFIN:RECord:IDS? query to get the beginning and end ID's of acquisition data records.

Returns `<span>,<sampling_freq>,<num_sample>,<center_freq>,<acq_BW>,<ref_level>,<trig_pos>,<time_stamp>,<FastFrame_ID>`

Where
`<span> ::` `<NR3>` is the span in Hz.
`<sampling_freq> ::` `<NR3>` is the sampling frequency in Hz.
`<num_sample> ::` `<NR3>` is the number of samples.
`<center_freq> ::` `<NR3>` is the center frequency in Hz.
`<acq_BW> ::` `<NR3>` is the acquisition bandwidth in Hz.
`<ref_level> ::` `<NR3>` is the reference level in dBm.
`<trig_pos> ::` `<NR3>` is the trigger position in seconds.
`<time_stamp> ::` `<string>` is the time stamp.
`<FastFrame_ID> ::` `<NR1>` is the Fast Frame ID: Zero or positive number. If the Fast Frame is disabled, the ID will be negative.

Examples `FETCH:RFIN:IQ:VHEADER? 10` might return
`110.0000000000E+6,150.0000000000E+6,4.0270000000E+3,2.7000000000E+9,110.0000000000E+6,0.0000,20.2420000000E-6,"7/9/2013 8:38:25.9305101 -07:00:00",-1`, indicating
Span: 110 MHz,
Sampling frequency: 150 MHz,
Number of samples: 4027,
Center frequency: 2.7 GHz,
Acquisition bandwidth: 110 MHz,
Reference level: 0 dBm,
Trigger position: 20.242 µs,
Time stamp: 10/31/2007 1118:32 AM,

Fast Frame ID: -1

FETCh:RFIN:RECord:IDS? (Query Only)

Returns the beginning and end ID numbers of acquisition data.

NOTE. *The instrument needs to be in stopped mode. If not in stopped mode, it returns the execution error (-200).*

Conditions Measurement views: All

Group Fetch commands

Syntax `FETCh:RFIN:RECord:IDS?`

Arguments None

Returns `<begin_ID>,<end_ID>`

Where
`<begin_ID> ::` `<NR1>` is the beginning ID of acquisition data.
`<end_ID> ::` `<NR1>` is the end ID of acquisition data.

NOTE. *"-1,-1" is returned when the span changes and the acquisition is armed, but the acquisition has not yet occurred.*

Examples `FETCH:RFIN:RECORD:IDS?` might return `1,147`, indicating the beginning and end ID's of acquisition data are 1 and 147, respectively.

FETCh:SEM:MHITs:COUNt? (Query Only)

Returns the number of rows in the results table in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Fetch commands

Syntax `FETCh:SEM:MHITs:COUNt?`

Arguments None

Returns <NRf> The number of rows in the table.

Examples `FETCH:SEM:MHITS:COUNT?` might return `7` indicating there are seven rows in the results table.

FETCh:SEM:MHITs<x>:AMPLitude:ABSolute? (Query Only)

Returns the absolute amplitude of the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Fetch commands

Syntax `FETCh:SEM:MHITs<x>:AMPLitude:ABSolute?`

Arguments None

Returns <NRf> Absolute amplitude of the specified signal in dBm.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples `FETCh:SEM:MHITs2:AMPLitude:ABSolute?` might return `–73.17` indicating that the absolute amplitude of signal in row 2 is –73.17 dBm.

FETCh:SEM:MHITs<x>:AMPLitude:RELative? (Query Only)

Returns the relative amplitude of the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Fetch commands

Syntax `FETCh:SEM:MHITs<x>:AMPLitude:RELative?`

Arguments None

Returns <NRf> Relative amplitude of the specified signal in dBm.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples `FETCh:SEM:MHITs2:AMPLitude:RELative?` might return `–91.81` indicating that the relative amplitude of mask hit signal in row 2 is –91.81 dBm.

FETCh:SEM:MHITs<x>:FREQuency? (Query Only)

Returns the frequency of the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Fetch commands

Syntax `FETCh:SEM:MHITs<x>:FREQuency?`

Arguments None

Returns <NRf> Relative amplitude of the specified signal in dBm.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples `FETCh:SEM:MHITs4:FREQuency?` might return `2.235E+6` indicating the frequency for mask hit signal in row 4 is 2.235 MHz.

FETCh:SEM:MHITs<x>:INTeg:ABSolute? (Query Only)

Returns the absolute integration amplitude of the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Fetch commands

Syntax `FETCh:SEM:MHITs<x>:INTeg:ABSolute?`

Arguments None

Returns `<NRf>` Absolute integration amplitude of the specified signal in dBm.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples `FETCh:SEM:MHITs4:INTeg:ABSolute?` might return `–75.14` indicating that the absolute integration amplitude for mask hit signal in row 4 is –75.14 dBM.

FETCh:SEM:MHITs<x>:INTeg:RELative? (Query Only)

Returns the relative integration amplitude of the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Fetch commands

Syntax `FETCh:SEM:MHITs<x>:INTeg:RELative?`

Arguments None

Returns `<NRf>` Relative integration amplitude of the specified signal in dBm.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples `FETCh:SEM:MHITs4:INTeg:RELative?` might return `–85.14` indicating that the relative integration amplitude for mask hit signal in row 4 is –85.14 dBM.

FETCh:SEM:MHITs<x>:MARGin:ABSolute? (Query Only)

Returns the absolute margin for the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Fetch commands

Syntax `FETCh:SEM:MHITs<x>:MARGin:ABSolute?`

Arguments None

Returns <NRf> Absolute margin of the specified signal in dB.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples `FETCh:SEM:MHITs4:MARGin:ABSolute?` might return `–62.33` indicating that the absolute margin for mask hit signal in row 4 is –62.33 dB.

FETCh:SEM:MHITs<x>:MARGin:RELative? (Query Only)

Returns the relative margin for the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Fetch commands

Syntax `FETCh:SEM:MHITs<x>:MARGin:RELative?`

Arguments None

Returns <NRf> Relative margin of the specified signal in dB.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples `FETCh:SEM:MHITs4:MARGin:RELative?` might return `–62.33` indicating that the relative margin for mask hit signal in row 4 is –62.33 dB.

FETCh:SEM:MHITs<x>:OFFSet? (Query Only)

Returns the offset for the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Fetch commands

Syntax `FETCh:SEM:MHITs<x>:OFFSet?`

Arguments None

Returns <NRf> specifies the mask hit offset.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples `FETCh:SEM:MHITs4:OFFSet?` might return `BL` indicating the offset for mask hit signal in row 4 is BL.

FETCh:SEM:PASS? (Query Only)

Returns the pass/fail limit test result in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Fetch commands

Syntax `FETCh:SEM:PASS?`

Arguments None

Returns 0 (fail) or 1 (pass)

Examples `FETCH:SEM:PASS?` might return 1, indicating that the limit test was successful.

FETCh:SEM:REF:POWer? (Query Only)

Returns the reference power level in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Fetch commands

Syntax `FETCh:SEM:REF:POWer?`

Arguments None

Returns `<NRf>` specifies the reference power.

Examples `FETCH:SEM:REF:POWER?` might return `-16.92 dBm` indicating that the reference power level is –16.92 dBm.

FETCh:SEM:SPECtrum:X? (Query Only)

Returns the frequencies of the spectrum trace in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Fetch commands

Syntax `FETCh:SEM:SPECtrum:X?`

Arguments None

Returns `#<num_digit><num_byte><x(1)><x(2)>...<x(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<x(n)>` is the frequency (Hz) at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:SEM:SPECTRUM:X?` might return `#516020xxxx...` (16020-byte data) for the frequencies of the spectrum trace.

FETCh:SEM:SPECtrum:XY? (Query Only)

Returns the frequency and amplitude pairs of the spectrum trace in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Fetch commands

Syntax `FETCh:SEM:SPECtrum:XY?`

Arguments None

Returns `#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<x(n)><y(n)>` is the frequency (Hz) and amplitude (dBm) pair at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:SEM:SPECTRUM:XY?` might return `#516020xxxx...` (16020-byte data) for the frequency and amplitude pairs of the spectrum trace.

FETCh:SEM:SPECtrum[:Y]? (Query Only)

Returns the amplitude of the spectrum trace in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Fetch commands

Syntax `FETCh:SEM:SPECtrum[:Y]?`

Arguments None

Returns `#<num_digit><num_byte><x(1)><x(2)>...<x(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<x(n)>` is the frequency (Hz) at the nth data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:SEM:SPECTRUM[:Y]?` might return `#516020xxxx...` (16020-byte data) for the frequencies of the spectrum trace.

FETCh:SGRam? (Query Only)

Returns trace data of a line in the spectrogram. The line is selected using the TRACe:SGRam:SELect:LINE command.

Conditions Measurement views: Spectrogram

Group Fetch commands

Syntax `FETCh:SGRam?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the amplitude in dBm at the nth data point,
4-byte little endian floating-point format specified in IEEE 488.2.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `FETCH:SGRAM?` might return `#43204xxxx...` (3204-byte data) for the line in the spectrogram.

FETCh:SPECtrum:TRACe<x>? (Query Only)

Returns the trace data in the Spectrum measurement.

The parameter <x> = 1 to 5.

NOTE. *TRACe5 (spectrogram) is valid when the spectrum and spectrogram measurements are running.*

Conditions Measurement views: Spectrum

Group Fetch commands

Syntax `FETCh:SPECtrum:TRACe<x>?`

Related Commands TRACe<x>:SPECtrum command subgroup

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the amplitude in dBm at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `FETCH:SPECTRUM:TRACE1?` might return `#43204xxxx...` (3204-byte data) for Trace 1 in the Spectrum measurement.

FETCh:SPURious:CARRier:POWer? (Query Only)

Returns the carrier power in the Spurious measurement.

Conditions Measurement views: Spurious

Group Fetch commands

Syntax `FETCh:SPURious:CARRier:POWer?`

Arguments None

Returns <NRf> Carrier power in dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `FETCH:SPURIOUS:CARRIER:POWER?` might return `4.227`, indicating that the carrier power is 4.227 dBm.

FETCh:SPURious:COUNt? (Query Only)

Returns the number of spurious signals in the Spurious measurement.

Conditions Measurement views: Spurious

Group Fetch commands

Syntax `FETCh:SPURious:COUNt?`

Arguments None

Returns <NRf> The number of spurious signals.

Examples `FETCH:SPURIOUS:COUNT?` might return `4`, indicating that the spurious count is 4.

FETCh:SPURious:PASS? (Query Only)

Returns the pass/fail limit test result in the Spurious measurement.

Conditions Measurement views: Spurious

Group Fetch commands

Syntax `FETCh:SPURious:PASS?`

Arguments None

Returns 0 (fail) or 1 (pass).

Examples FETCH:SPURIOUS:PASS? might return 1, indicating that the limit test was successful.

FETCh:SPURious:SPECtrum:X? (Query Only)

Returns the frequencies of the spectrum trace in the Spurious measurement.

Conditions Measurement views: Spurious

Group Fetch commands

Syntax FETCh:SPURious:SPECtrum:X?

Arguments None

Returns #<num_digit><num_byte><x(1)><x(2)>...<x(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<x(n)> is the frequency (Hz) at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples FETCH:SPURIOUS:SPECTRUM:X? might return #516020xxxx... (16020-byte data) for the frequencies of the spectrum trace in the Spurious measurement.

FETCh:SPURious:SPECtrum:XY? (Query Only)

Returns the frequency and amplitude pairs of the spectrum trace in the Spurious measurement.

Conditions Measurement views: Spurious

Group Fetch commands

Syntax `FETCh:SPURious:SPECtrum:XY?`

Arguments None

Returns `#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<x(n)><y(n)>` is the frequency (Hz) and amplitude (dBm) pair at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.
The amplitude unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `FETCH:SPURIOUS:SPECTRUM:XY?` might return `#516020xxxx...` (16020-byte data) for the frequency and amplitude pairs of the spectrum trace in the Spurious measurement.

FETCh:SPURious:SPECtrum[:Y]? (Query Only)

Returns the amplitudes of the spectrum trace in the Spurious measurement.

Conditions Measurement views: Spurious

Group Fetch commands

Syntax `FETCh:SPURious:SPECtrum[:Y]?`

Arguments None

Returns `#<num_digit><num_byte><y(1)><y(2)>...<y(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<y(n)>` is the amplitude (dBm) at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `FETCH:SPURIOUS:SPECTRUM:Y?` might return `#516020xxxx...` (16020-byte data) for the amplitudes of the spectrum trace in the Spurious measurement.

FETCh:SPURious:SPUR<x>:AMPLitude:ABSolute? (Query Only)

Returns the absolute amplitude of the specified spurious signal in the Spurious measurement.

Conditions Measurement views: Spurious

Group Fetch commands

Syntax `FETCh:SPURious:SPUR<x>:AMPLitude:ABSolute?`

Arguments None

Returns <NRf> Absolute amplitude of the specified spurious signal in dBm. The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `FETCH:SPURIOUS:SPUR1:AMPLITUDE:ABSOLUTE?` might return `-19.782`, indicating that the absolute amplitude of Spurious #1 is -19.782 dBm.

FETCh:SPURious:SPUR<x>:AMPLitude:RELative? (Query Only)

Returns the relative amplitude of the specified spurious signal in the Spurious measurement.

Conditions Measurement views: Spurious

Group Fetch commands

Syntax `FETCh:SPURious:SPUR<x>:AMPLitude:RELative?`

Arguments None

Returns <NRf> Relative amplitude of the specified spurious signal in dB. Use the [SENSe]:SPURious:REFerence command to set the power reference.

Examples `FETCH:SPURIOUS:SPUR1:AMPLITUDE:RELATIVE?` might return `-9.782`, indicating that the relative amplitude of Spurious #1 is -9.782 dB.

FETCh:SPURious:SPUR<x>:FREQuency:ABSolute? (Query Only)

Returns the absolute frequency of the specified spurious signal in the Spurious measurement.

Conditions Measurement views: Spurious

Group Fetch commands

Syntax `FETCh:SPURious:SPUR<x>:FREQuency:ABSolute?`

Arguments None

Returns <NRf> Absolute frequency of the spurious signal in Hz.

Examples `FETCH:SPURIOUS:SPUR1:FREQUENCY:ABSOLUTE?` might return `2.235E+9`, indicating that the absolute frequency of Spurious #1 is 2.235 GHz.

FETCh:SPURious:SPUR<x>:FREQuency:RELative? (Query Only)

Returns the relative frequency of the specified spurious signal to the carrier in the Spurious measurement. This command is valid when [SENSe]:SPURious:REFerence is set to CARRier.

Conditions Measurement views: Spurious

Group Fetch commands

Syntax `FETCh:SPURious:SPUR<x>:FREQuency:RELative?`

Arguments None

Returns <NRf> Relative frequency of the spurious signal to the carrier in Hz.

Examples `FETCH:SPURIOUS:SPUR1:FREQUENCY:RELATIVE` might return `3.634E+6`, indicating that the relative frequency of Spurious #1 is 3.634 MHz.

FETCh:SPURious:SPUR<x>:LIMit:ABSolute? (Query Only)

Returns the absolute amplitude of the limit for the specified spurious signal in the Spurious measurement.

Conditions Measurement views: Spurious

Group Fetch commands

Syntax `FETCh:SPURious:SPUR<x>:LIMit:ABSolute?`

Arguments None

Returns `<NRf>` Absolute amplitude of the limit for the specified spurious signal in dBm. The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `FETCH:SPURIOUS:SPUR1:LIMIT:ABSOLUTE?` might return `-50.0`, indicating that the absolute amplitude of the limit for Spurious #1 is -50 dBm.

FETCh:SPURious:SPUR<x>:LIMit:RELative? (Query Only)

Returns the relative amplitude of the limit for the specified spurious signal in the Spurious measurement.

Conditions Measurement views: Spurious

Group Fetch commands

Syntax `FETCh:SPURious:SPUR<x>:LIMit:RELative?`

Arguments None

Returns `<NRf>` Relative amplitude of the limit for the specified spurious signal in dB. Use the [SENSe]:SPURious:REFerence command to set the power reference.

Examples `FETCH:SPURIOUS:SPUR1:LIMIT:RELATIVE?` might return `-10.0`, indicating that the relative amplitude of the limit for Spurious #1 is -10 dB.

FETCh:SPURious:SPUR<x>:LIMit:VIOLation? (Query Only)

Returns whether the specified spurious signal exceeds the limit or not.

Conditions Measurement views: Spurious

Group Fetch commands

Syntax `FETCh:SPURious:SPUR<x>:LIMit:VIOLation?`

Arguments None

Returns `0` (under the limit) or `1` (over the limit).

Examples `FETCH:SPURIOUS:SPUR1:LIMIT:VIOLATION?` might return `1`, indicating that Spurious #1 exceeds the limit.

FETCh:SPURious:SPUR<x>:RANGe? (Query Only)

Returns the frequency range in which the specified spurious signal occurred.

Conditions Measurement views: Spurious

Group Fetch commands

Syntax `FETCh:SPURious:SPUR<x>:RANGe?`

Arguments None

Returns `<string>` "A" to "T" representing Range A to T, respectively.

Examples `FETCH:SPURIOUS:SPUR1:RANGE` might return `"E"`, indicating that Spurious #1 is in Range E.

FETCh:SQUality:FREQuency:DEViation? (Query Only)

Returns the frequency deviation in the signal quality measurement.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to C4FM, FSK2, FSK4, FSK8, or FSK16.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax `FETCh:SQUality:FREQuency:DEViation?`

Arguments None

Returns `<NRf>` Frequency deviation in Hz.

Examples `FETCH:SQUALITY:FREQUENCY:DEVIATION?` might return `12.68E+3`, indicating the frequency deviation is 12.68 kHz.

FETCh:SQUality:FREQuency:DEViation:TABLe? (Query Only)

Returns the number of columns and the values in the frequency deviation table for a signal quality measurement.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to C4FM, FSK2, FSK4, FSK8, or FSK16.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax `FETCh:SQUality:FREQuency:DEViation:TABLe?`

Arguments None

Returns `<Dev_Num>,{<Freq_dev_Max>,<Freq_dev_Min>,<Freq_dev_Avg>}`

Where

<Dev_Num> :: <NR1> is the number of columns in the frequency deviation table.
= 2 (2FSK), 4 (4FSK, C4FM), 8 (8FSK), or 16 (16FSK)
<Freq_dev_Max> :: <NRf> is the maximum frequency deviation in Hz.
<Freq_dev_Min> :: <NRf> is the minimum frequency deviation in Hz.
<Freq_dev_Avg> :: <NRf> is the average frequency deviation in Hz.

The dataset <Freq_dev_Max>,<Freq_dev_Min>,<Freq_dev_Avg> is returned for each symbol in ascending order of its level (for example, in order of symbol -3, -1, +1, and +3 for 4FSK).

Examples FETCH:SQUALITY:FREQUENCY:DEVIATION:TABLE? might return 2,1.257E+3,1.039E+3,1.162E+3,1.586E+3,1.298E+3,1.425E+3 for the frequency signal, populating the results table as follows.

Deviations	-1	+1
Maximum	1.257 kHz	1.586 kHz
Minimum	1.039 kHz	1.298 kHz
Average	1.162 kHz	1.425 kHz

FETCh:SQUality:FREQuency:ERRor? (Query Only)

Returns the frequency error in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax FETCh:SQUality:FREQuency:ERRor?

Arguments None

Returns <NRf> Frequency error in Hz.

Examples FETCH:SQUALITY:FREQUENCY:ERROR? might return 612.043E+3, indicating that the frequency error is 612.043 kHz.

FETCh:SQUality:GAIN:IMBalance? (Query Only)

Returns the gain imbalance in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax `FETCh:SQUality:GAIN:IMBalance?`

Arguments None

Returns <NRf> Gain imbalance in dB.

Examples `FETCH:SQUALITY:GAIN:IMBALANCE?` might return `-57.746E-3`, indicating that the gain imbalance is -0.057746 dB.

FETCh:SQUality:ORIGin:OFFSet? (Query Only)

Returns the origin offset in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax `FETCh:SQUality:ORIGin:OFFSet?`

Arguments None

Returns <NRf> Origin offset in dB.

Examples `FETCH:SQUALITY:ORIGIN:OFFSET?` might return `-44.968`, indicating that the origin offset is -44.968 dB.

FETCh:SQUality:PEAK:EVM? (Query Only)

Returns the peak EVM (%) in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax `FETCh:SQUality:PEAK:EVM?`

Arguments None

Returns <NRf> Peak EVM in percent (%).

Examples `FETCH:SQUALITY:PEAK:EVM?` might return `4.276`, indicating that the peak EVM is 4.276%.

FETCh:SQUality:PEAK:EVM:DB? (Query Only)

Returns the peak EVM (dB) in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax `FETCh:SQUality:PEAK:EVM:DB?`

Arguments None

Returns <NRf> Peak EVM in dB.

Examples `FETCH:SQUALITY:PEAK:EVM:DB?` might return `-27.358`, indicating that the peak EVM is -27.358 dB.

FETCh:SQUality:PEAK:EVM:DB:OFFSet? (Query Only)

Returns the peak offset EVM (dB) in the signal quality measurement.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to OQPSK or SOQPSK.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax `FETCh:SQUality:PEAK:EVM:DB:OFFSet?`

Arguments None

Returns `<NRf>` Peak offset EVM in dB.

Examples `FETCH:SQUALITY:PEAK:EVM:DB:OFFSET?` might return `-37.624`, indicating the peak offset EVM is -37.624 dB.

FETCh:SQUality:PEAK:EVM:LOCation? (Query Only)

Returns the time at which the EVM is peak.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax `FETCh:SQUality:PEAK:EVM:LOCation?`

Arguments None

Returns `<NRf>` The time in symbol number at which the EVM is peak.
The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples `FETCH:SQUALITY:PEAK:EVM:LOCATION?` might return `68.000`, indicating that the EVM is peak at symbol #68.000.

FETCh:SQUality:PEAK:EVM:LOCation:OFFSet? (Query Only)

Returns the time at which the offset EVM is peak.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to OQPSK or SOQPSK.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax `FETCh:SQUality:PEAK:EVM:LOCation:OFFSet?`

Arguments None

Returns <NRf> The time in symbol number at which the offset EVM is peak. The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples `FETCH:SQUALITY:PEAK:EVM:LOCATION:OFFSET?` might return `123.00`, indicating that the offset EVM is peak at symbol #123.

FETCh:SQUality:PEAK:EVM:OFFSet? (Query Only)

Returns the peak offset EVM (%) in the signal quality measurement.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to OQPSK or SOQPSK.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax `FETCh:SQUality:PEAK:EVM:OFFSet?`

Arguments None

Returns <NRf> Peak offset EVM in percent (%).

Examples `FETCH:SQUALITY:PEAK:EVM:OFFSET?` might return `1.298`, indicating the peak offset EVM is 1.298%.

FETCh:SQUality:PEAK:FERRor? (Query Only)

Returns the peak FSK error in the signal quality measurement.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to FSK2, FSK4, FSK8, or FSK16.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax `FETCh:SQUality:PEAK:FERRor?`

Arguments None

Returns <NRf> Peak FSK error in percent (%).

Examples `FETCH:SQUALITY:PEAK:FERROR?` might return `9.136`, indicating the peak FSK error is 9.136%.

FETCh:SQUality:PEAK:MERRor? (Query Only)

Returns the peak magnitude error (%) in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax `FETCh:SQUality:PEAK:MERRor?`

Arguments None

Returns <NRf> Peak magnitude error in percent (%).

Examples `FETCH:SQUALITY:PEAK:MERROR?` might return `3.595`, indicating that the peak magnitude error is 3.595%.

FETCh:SQUality:PEAK:MERRor:DB? (Query Only)

Returns the peak magnitude error (dB) in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax `FETCh:SQUality:PEAK:MERRor:DB?`

Arguments None

Returns `<NRf>` Peak magnitude error in dB.

Examples `FETCH:SQUALITY:PEAK:MERROR:DB?` might return `-28.583`, indicating that the magnitude error is -28.583 dB.

FETCh:SQUality:PEAK:MERRor:LOCation? (Query Only)

Returns the time at which the magnitude error is peak.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax `FETCh:SQUality:PEAK:MERRor:LOCation?`

Arguments None

Returns `<NRf>` The time in symbol number at which the magnitude error is peak. The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples `FETCH:SQUALITY:PEAK:MERROR:LOCATION?` might return `68.000`, indicating that the magnitude error is peak at symbol #68.

FETCh:SQUality:PEAK:PERRor? (Query Only)

Returns the peak phase error in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax `FETCh:SQUality:PEAK:PERRor?`

Arguments None

Returns <NRf> Peak phase error in degrees.

Examples `FETCH:SQUALITY:PEAK:PERROR?` might return `1.907`, indicating that the peak phase error is 1.907 °.

FETCh:SQUality:PEAK:PERRor:LOCation? (Query Only)

Returns the time at which the phase error is peak.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax `FETCh:SQUality:PEAK:PERRor:LOCation?`

Arguments None

Returns <NRf> The time in symbol number at which the phase error is peak.
The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples `FETCH:SQUALITY:PEAK:PERROR:LOCATION?` might return `68.000`, indicating that the phase error is peak at symbol #68.

FETCh:SQUality:QUADrature:ERRor? (Query Only)

Returns the quadrature error in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax `FETCh:SQUality:QUADrature:ERRor?`

Arguments None

Returns <NRf> Quadrature error in degrees.

Examples FETCH:SQUALITY:QUADRATURE:ERROR? might return -14.264E-3, indicating that the quadrature error is -0.014264°.

FETCh:SQUality:RHO? (Query Only)

Returns the ρ (waveform quality) value in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax FETCh:SQUality:RHO?

Arguments None

Returns <NRf> ρ value.

Examples FETCH:SQUALITY:RHO? might return 998.703E-3, indicating that ρ is 0.998703.

FETCh:SQUality:RMS:EVM? (Query Only)

Returns the RMS EVM (%) in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax FETCh:SQUality:RMS:EVM?

Arguments None

Returns <NRf> RMS EVM in percent (%).

Examples FETCH:SQUALITY:RMS:EVM? might return 2.417, indicating that the RMS EVM is 2.417%.

FETCh:SQUality:RMS:EVM:DB? (Query Only)

Returns the RMS EVM (dB) in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax FETCh:SQUality:RMS:EVM:DB?

Arguments None

Returns <NRf> RMS EVM in dB.

Examples FETCH:SQUALITY:RMS:EVM:DB? might return -32.356, indicating that the RMS EVM is -32.356 dB.

FETCh:SQUality:RMS:EVM:DB:OFFSet? (Query Only)

Returns the RMS offset EVM (dB) in the signal quality measurement.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to OQPSK or SOQPSK.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax FETCh:SQUality:RMS:EVM:DB:OFFSet?

Arguments None

Returns <NRf> RMS offset EVM in dB.

Examples FETCH:SQUALITY:RMS:EVM:DB:OFFSET? might return –41.276, indicating the RMS offset EVM is -41.276 dB.

FETCh:SQUality:RMS:EVM:OFFSet? (Query Only)

Returns the RMS offset EVM (%) in the signal quality measurement.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to OQPSK or SOQPSK.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax FETCh:SQUality:RMS:EVM:OFFSet?

Arguments None

Returns <NRf> RMS offset EVM in percent (%).

Examples FETCH:SQUALITY:RMS:EVM:OFFSET? might return 0.783, indicating the RMS offset EVM is 0.783%.

FETCh:SQUality:RMS:FERRor? (Query Only)

Returns the RMS FSK error percent in the signal quality measurement.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to FSK2, FSK4, FSK8, or FSK16.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax FETCh:SQUality:RMS:FERRor?

Arguments None

Returns <NRf> RMS FSK error in percent (%).

Examples FETCH:SQUALITY:RMS:FERROR? might return 1.815, indicating the RMS FSK error is 1.815%.

FETCh:SQUality:RMS:MER:DB? (Query Only)

Returns the RMS MER (Modulation Error Ratio) in dB in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax FETCh:SQUality:RMS:MER:DB?

Arguments None

Returns <NRf> RMS MER in dB.

Examples FETCH:SQUALITY:RMS:MER:DB? might return 27.394, indicating that the RMS MER is 27.394 dB.

FETCh:SQUality:RMS:MERRor? (Query Only)

Returns the RMS magnitude error (%) in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax FETCh:SQUality:RMS:MERRor?

Arguments None

Returns <NRf> RMS magnitude error in percent (%).

Examples FETCH:SQUALITY:RMS:MERROR? might return 1.837, indicating that the RMS magnitude error is 1.837%.

FETCh:SQUality:RMS:MERRor:DB? (Query Only)

Returns the RMS magnitude error (dB) in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax FETCh:SQUality:RMS:MERRor:DB?

Arguments None

Returns <NRf> RMS MERRor in dB.

Examples FETCH:SQUALITY:RMS:MERROR:DB? might return -34.706, indicating that the magnitude error is -34.706 dB.

FETCh:SQUality:RMS:PERRor? (Query Only)

Returns the RMS phase error in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Fetch commands

Syntax FETCh:SQUality:RMS:PERRor?

Arguments None

Returns <NRf> RMS phase error in degrees.

Examples `FETCH:SQUALITY:RMS:PERROR?` might return `893.472E-3`, indicating that the RMS phase error is 0.893472 °.

FETCh:SQUality:SYMBol:LENGth? (Query Only)

Returns the number of analyzed symbols.

Group Fetch commands

Syntax `FETCh:SQUality:SYMBol:LENGth?`

Related Commands

Returns `<NR1>` indicates the length of the synch word in symbols.

Examples `FETCH:SQUALITY:SYMBOL:LENGTH?`might return `3`, indicating the length is three symbols.

FETCh:SQUality:SYMBol:RATE? (Query Only)

Returns the value of the calculated symbol rate in Hz.

Conditions Measurement views: Signal quality

It is valid when the modulation type is 2|4|8|16FSK and [:SENSe]:DDEMod:SYMBol:RATE:SEARch is ON.

Group Fetch commands

Syntax `FETCh:SQUality:SYMBol:RATE?`

Related Commands [SENSe]:DDEMod:SYMBol:RATE:SEARch

Returns `<NRf>` is the calculated symbol rate in Hz.

Examples `FETCH:SQUALITY:SYMBOL:RATE?` might return `95.24E+3`, indicating the calculated symbol rate is 95.24 kHz.

FETCh:SQUality:SYMBol:RATE:ERRor? (Query Only)

Returns the value of the symbol rate error in percent (%).

Conditions Measurement views: Signal quality

It is valid when the modulation type is 2|4|8|16FSK and [:SENSe]:DDEMod:SYMBol:RATE:SEARch is ON.

Group Fetch commands

Syntax `FETCh:SQUality:SYMBol:RATE:ERRor?`

Related Commands [SENSe]:DDEMod:SYMBol:RATE:SEARch

Returns <NRf> is the symbol error percent (%).

Examples `FETCH:SQUALITY:SYMBOL:RATE:ERROR?` might return `-0.002`, indicating that the symbol rate error is -0.002%.

FETCh:TDIagram:FERRor? (Query Only)

Returns the frequency error in the trellis diagram measurement.

Conditions Measurement views: Trellis diagram

Group Fetch commands

Syntax `FETCh:TDIagram:FERRor?`

Arguments None

Returns <freq_error> :: <NRf> is the frequency error in Hz.

Examples `FETCH:TDIAGRAM:FERROR?` might return `-10.7E+3`, indicating the frequency error is -10.7 kHz.

FETCh:TDIagram:TRACe? (Query Only)

Returns the Trellis diagram trace data.

Conditions Measurement views: Trellis diagram

Group Fetch commands

Syntax `FETCh:TDIagram:TRACe?`

Arguments None

Returns `#<num_digit><num_byte><Y(1)><X(1)><Y(2)><X(2)>...<Y(n)><X(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<Y(n)><X(n)>` is the phase in degrees and the time in symbols pair at the n^{th} data point. The 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:TDIAGRAM:TRACE?` might return `#3160xxxx...` (160-byte data) for the Trellis diagram trace.

FETCh:TOVerview? (Query Only)

Returns the trace data in the time overview.

Conditions Measurement views: Time overview

Group Fetch commands

Syntax `FETCh:TOVerview?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.
<data(n)> is the amplitude in dBm at the nth data point,
4-byte little endian floating-point format specified in IEEE 488.2.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples FETCH:TOVERVIEW? might return #43204xxxx... (3204-byte data) for the trace in the time overview.

FETCh:TXGain:AVERage? (Query Only)

Returns the mean value of the trace.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax FETCh:TXGain:AVERage?

Returns <NRf> is the averaged value of the trace points in dB.

Examples FETCH:TXGAIN:AVERAGE? might return 2.3, indicating that the mean value of the trace is 2.3 dB.

FETCh:TXGain:MAX? (Query Only)

Returns the maximum trace value.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax FETCh:TXGain:MAX?

Returns <NRf> is the maximum value of the trace in dB.

Examples FETCH:TXGAIN:MAX? might return -3.45, indicating the maximum trace value is -3.45 dB.

FETCh:TXGain:MAXLocation? (Query Only)

Returns the frequency of the maximum trace value.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax `FETCh:TXGain:MAXLocation?`

Returns <NRf> is the frequency at the maximum trace value.

Examples `FETCH:TXGAIN:MAXLOCATION?` might return 1E9, indicating the maximum trace value is at 1 GHz.

FETCh:TXGain:MIN? (Query Only)

Returns the minimum trace value.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax `FETCh:TXGain:MIN?`

Returns <NRf> is the frequency at the minimum trace value in dB.

Examples `FETCH:TXGAIN:MIN?` might return -30, indicating the minimum trace value is -30 dB.

FETCh:TXGain:MINLocation? (Query Only)

Returns the frequency of the minimum trace value.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax `FETCh:TXGain:MINLocation?`

Returns <NRf> is the frequency at the minimum trace value.

Examples `FETCH:TXGAIN:MINLOCATION?` might return 1E6, indicating the minimum trace value is at 1 MHz.

FETCh:TXGain:TRACe<x>? (Query Only)

Returns the trace data of the selected trace.

Conditions Measurement view: Transmission Gain

Where <x> is the trace 1, 2, or 3.

Group Display group

Syntax `FETCh:TXGain:TRACe<x>?`

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the amplitude in dB at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCh:TXGain:TRACe1?` might return #43204xxxx... (3204 byte data) for Trace 1 in the measurement.

FETCh:WLAN:CONSte:MAGNitude? (Query Only)

Returns the magnitude data for the WLAN Constellation measurement.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:CONSte:MAGNitude?`

Returns Returned data is in the form

#<num_digit><num_byte><data(1)><data(2)>..<data(n)>

where

<num_digit> is the number of digits in <num_byte>

<num_byte> is the number of bytes of data that follow

<data(n)> is the data value at the nth data point in dB, 4-byte little endian floating-point format specified in IEEE 488.2

Examples `FETC:WLAN:CONS:MAGN?` might return `#510400..` (10400-byte data) for the measurement.

FETCh:WLAN:CONSte:PHASe? (Query Only)

Returns the phase values for the WLAN Constellation measurement.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:CONSte:PHASe?`

Returns Returned data is in the form

#<num_digit><num_byte><data(1)><data(2)>..<data(n)>

where

- <num_digit> is the number of digits in <num_byte>
- <num_byte> is the number of bytes of data that follow
- <data(n)> is the data value at the nth data point in dB, 4-byte little endian floating-point format specified in IEEE 488.2

Examples `FETCH:WLAN:CONS:PHAS?` might return `#510400..` (10400-byte data) for the measurement.

FETCh:WLAN:CONSte:TYPE? (Query Only)

Returns the type (context) for the WLAN Constellation measurement.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:CONSte:TYPE?`

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in 16-bit little endian integers. The integers must be translated to the proper context as listed below.

Integer value Type (context):

- 1 Pilot
- 2 Data

Examples `FETCH:WLAN:CONS:TYPE?` might return `#41352..` (1352-byte data) for the measurement. The actual data must be decoded to the context type using the table above.

FETCh:WLAN:CONSte:VALue? (Query Only)

Returns the value (decoded symbol) for the WLAN Constellation measurement.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:CONSte:VALue?`

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in 16-bit little endian integers.

Examples `FETCH:WLAN:CONSTE:VALUE?` might return `#41352` indicating 1352 bytes of data for the measurement.

FETCh:WLAN:CRESponse:MAGNitude? (Query Only)

Returns the magnitude data of the WLAN Channel Response measurement.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:CRESponse:MAGNitude?`

Returns Returned data is in the form

#<num_digit><num_byte><data(1)><data(2)>..<data(n)>

where

<num_digit> is the number of digits in <num_byte>

<num_byte> is the number of bytes of data that follow

<data(n)> is the data value at the nth data point in dB, 4-byte little endian floating-point format specified in IEEE 488.2

Examples `FETC:WLAN:CRES:MAGN?` might return `#510400..` (10400-byte data) for the measurement.

FETCh:WLAN:CRESponse:PHASe? (Query Only)

Returns the phase data of the WLAN Channel Response measurement.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:CRESponse:PHASe?`

Returns Returned data is in the form

#<num_digit><num_byte><data(1)><data(2)>..<data(n)>

where

- <num_digit> is the number of digits in <num_byte>
- <num_byte> is the number of bytes of data that follow
- <data(n)> is the data value at the nth data point in dB, 4-byte little endian floating-point format specified in IEEE 488.2

Examples `FETCH:WLAN:CRES:PHAS?` might return `#510400..` (10400-byte data) representing the phase data for the measurement.

FETCh:WLAN:EVM:TRACe<x>? (Query Only)

Returns the trace data for the selected WLAN EVM trace.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbol.

When <x> is 3, the parameter is Average versus Subcarrier

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:EVM:TRACe<x>?`

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where:

<num_digit> is the number of digits in <num_byte.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:WLAN:EVM:TRAC2?` might return `#510400..` (10400-byte data) for the measurement.

FETCh:WLAN:FLATness:PASS? (Query Only)

Queries whether the average power level values of the subcarriers across the signal bandwidth on the WLAN Spectral Flatness display remain within the limits defined for a particular standard.

NOTE. *WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:FLATness:PASS?`

Returns Returns "1" in the case that all data remains within the defined limits, and "0" otherwise.

Examples `FETCH:WLAN:FLAT:PASS?` might return 1, which means that the average power level values of the subcarriers across the signal bandwidth remain within the defined limits.

FETCh:WLAN:FLATness:RESult? (Query Only)

Returns the summary results of each segment of the WLAN Spectral Flatness display.

NOTE. *WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:FLATness:RESult?`

Returns The data is returned as 12 comma delineated text values. The specific values are :

- minimum subcarrier within segment 1
- minimum subcarrier average energy deviation from the global average within segment 1
- deviation the of minimum subcarrier average from the allowed value in segment 1
- maximum subcarrier within segment 1
- maximum subcarrier average energy deviation from the global average within segment 1
- deviation of the maximum subcarrier average from the allowed value in segment 1
- minimum subcarrier within segment 2
- minimum subcarrier average energy deviation from the global average within segment 2
- deviation the of minimum subcarrier average from the allowed value in segment 2
- maximum subcarrier within segment 2
- maximum subcarrier average energy deviation from the global average within segment 2
- deviation of the maximum subcarrier average from the allowed value in segment 2

Examples `FETCH:WLAN:FLAT:RES?` might return `-6.0000000000,-17.0626174659E-3,3.9829373825,12.0000000000,19.97926086` which represents the summary results of each segment of the WLAN Spectral Flatness display.

FETCh:WLAN:FLATness:TRACe<x>? (Query Only)

Returns the WLAN Spectral Flatness trace data. When <x> is 1, the parameter is Matrix (symbol deviation per subcarrier). When<x> is 2, the parameter is Average Deviation vs Subcarrier.

NOTE. *WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:FLATness:TRACe<x>?`

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where:

<num_digit> is the number of digits in <num_byte.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:WLAN:FLAT:TRACE2?` might return `#510400..` (10400-byte data) for the measurement.

FETCh:WLAN:MERRor:TRACe<x>? (Query Only)

Returns the trace data for the selected WLAN Magnitude Error trace.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbol.

When <x> is 3, the parameter is Average versus Subcarrier

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:MERRor:TRACe<x>?`

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where:

<num_digit> is the number of digits in <num_byte.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian

floating-point format specified in IEEE 488.2.

Examples `FETCH:WLAN:MERR:TRAC2?` might return `#510400..` (10400-byte data) for the measurement.

FETCh:WLAN:PERRor:TRACe<x>? (Query Only)

Returns the trace data for the selected WLAN Phase Error trace.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbol.

When <x> is 3, the parameter is Average versus Subcarrier

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:PERRor:TRACe<x>?`

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where:

<num_digit> is the number of digits in <num_byte.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:WLAN:PERR:TRAC2?` might return `#510400..` (10400-byte data) for the measurement.

FETCh:WLAN:PVTime:BURSt:POWer? (Query Only)

Returns the average power of the burst packet, in dBm, for the WLAN Power vs. Time trace.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:PVTime:BURSt:POWer?`

Returns Floating point number that represents the average power of the burst packet, in dBm.

Examples `FETCH:WLAN:PVT:BURST:POWER?` might return `-1.99`, which represents the value of the average power of the burst packet of -1.99 dBm.

FETCh:WLAN:PVTime:BURSt:WIDTh? (Query Only)

Returns the value of the measured time width of the burst packet from Power-On to Power-Down (or end of waveform, if that occurs before Power-Down), in seconds, for the WLAN Power vs. Time trace.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:PVTime:BURSt:WIDTh?`

Returns Floating point number which represents the value of the measured time width of the burst packet from Power-On to Power-Down (or end of waveform, if that occurs before Power-Down), in seconds.

Examples `FETCH:WLAN:PVT:BURST:WIDTH?` might return `156.0000000000E-6`, which represents a width of the burst packet of 156 µS.

FETCh:WLAN:PVTime:FALL:TIME? (Query Only)

Returns the value of the width of the falling edge, in seconds.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:PVTime:FALL:TIME?`

Returns Floating point number which represents the value of the width of the falling edge of the burst, in seconds.

Examples `FETCH:WLAN:PVT:FALL:TIME?` might return `7e-6`, which indicates the width of the falling edge is 7 µs.

FETCh:WLAN:PVTime:FALL:TIME:STARt? (Query Only)

Returns the value of the start time of the falling edge, in seconds.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:PVTime:FALL:TIME:STARt?`

Returns Floating point number which represents the value of the start time of the falling edge of the burst, in seconds.

Examples `FETCH:WLAN:PVT:FALL:TIME:STAR?` might return `6e-6`, which indicates that the start time of the falling edge is at 6 µs.

FETCh:WLAN:PVTime:FALL:TIME:STOP? (Query Only)

Returns the value of the stop time of the falling edge, in seconds.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:PVTime:FALL:TIME:STOP?`

Returns Floating point number which represents the value of the stop time of the falling edge of the burst, in seconds.

Examples `FETCH:WLAN:PVT:FALL:TIME:STOP?` might return `7e-6`, which indicates the stop time of the falling edge is at 7 µs.

FETCh:WLAN:PVTime:FTIMe? (Query Only)

Returns the value of the width of the falling edge, in seconds.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:PVTime:FTIMe?`

Returns Floating point number which represents the value of the width of the falling edge of the burst, in seconds.

Examples `FETCH:WLAN:PVT:FALL:TIME?` might return `7e-6`, which indicates the width of the falling edge is 7 µs.

FETCh:WLAN:PVTime:RISE:TIME? (Query Only)

Returns the value of the width of the rising edge, in seconds.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:PVTime:RISE:TIME?`

Returns Floating point number which represents the value of the width of the rising edge of the burst, in seconds.

Examples `FETCH:WLAN:PVT:RISE:TIME?` might return `7e-6`, which indicates the width of the rising edge is 7 µs.

FETCh:WLAN:PVTime:RISE:TIME:STARt? (Query Only)

Returns the value of the start time of the rising edge, in seconds.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:PVTime:RISE:TIME:STARt?`

Returns Floating point number which represents the value of the start time of the rising edge of the burst, in seconds.

Examples `FETCH:WLAN:PVT:RISE:TIME:STAR?` might return `6e-6`, which indicates the start time of the rising edge is at 6 µs.

FETCh:WLAN:PVTime:RISE:TIME:STOP? (Query Only)

Returns the value of the stop time of the rising edge, in seconds.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:PVTime:RISE:TIME:STOP?`

Returns Floating point number which represents the value of the stop time of the rising edge of the burst, in seconds.

Examples `FETCH:WLAN:PVT:RISE:TIME:STOP?` might return `7e-6`, which indicates the stop time of the rising edge is at 7µs.

FETCh:WLAN:PVTime:RTIMe? (Query Only)

Returns the value of the width of the rising edge, in seconds.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:PVTime:RTIMe?`

Returns Floating point number which represents the value of the width of the rising edge of the burst, in seconds.

Examples `FETCH:WLAN:PVT:RISE:TIME?` might return `7e-6`, which indicates the width of the rising edge is 7 µs.

FETCh:WLAN:PVTime:TRACe:X? (Query Only)

Returns the horizontal values (time in seconds) for the WLAN Power vs. Time trace.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:PVTime:TRACe:X?`

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where:

<num_digit> the number of digits in <num_byte.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:WLAN:PVT:TRAC:X` might return `#43204` (3204-byte data), which represents the time in seconds (horizontal values) for the WLAN Power vs. Time trace.

FETCh:WLAN:PVTime:TRACe:XY? (Query Only)

Returns the horizontal value (time in seconds) and vertical value (power) for the WLAN Power vs. Time trace.

<x(n)><y(n)> is the horizontal value and vertical value pair at the nth data point.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:PVTime:TRACe:XY?`

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where:

<num_digit> is the number of digits in <num_byte.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:WLAN:PVT:TRAC:XY?` might return `#43204` (3204-byte data), which represents the horizontal value (time in seconds) and vertical value (power) pair at the nth data point.

FETCh:WLAN:PVTime:TRACe[:Y]? (Query Only)

Returns the vertical values (power) for the WLAN Power vs. Time trace.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:PVTime:TRACe[:Y]?`

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where:

<num_digit> is the number of digits in <num_byte.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:WLAN:PVT:TRAC:Y?` might return `#43204` (3204-byte data), which represents the vertical values (power) for the WLAN Power vs. Time trace.

FETCh:WLAN:STABle:VALUe? (Query Only)

Returns the number of bytes in the symbol table for the WLAN measurement.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:STABle:VALUe?`

Returns #<num_digit><num_byte>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

Examples `FETCH:WLAN:STABLE:VALUE?` might return `#43848` indicating that there are 3848 bytes of data in the symbol table.

FETCh:WLAN:SUMMary:BURSt:INDex? (Query Only)

Returns the index of the analyzed packet within the analysis record.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:BURSt:INDex?`

Arguments None

Returns <NRf>

FETCh:WLAN:SUMMary:BURSt:POWer? (Query Only)

Returns the average power of all symbols in the packet, including Preamble and Data segments.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:BURSt:POWer?`

Arguments None

Returns <NRf> data in dBm.

FETCh:WLAN:SUMMary:BURSt:POWer:CFACtor? (Query Only)

Returns the peak-to-average burst power factor in the WLAN measurement. The ratio of the highest instantaneous signal power level to the average signal power.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:BURSt:POWer:CFACtor?`

Arguments None

Returns <NRf> data in dB.

FETCh:WLAN:SUMMary:CERRor? (Query Only)

Returns the symbol clock error in the WLAN measurement.

Not available for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:CERRor?`

Arguments None

Returns <NRf> data in parts per million.

FETCh:WLAN:SUMMary:CPE? (Query Only)

Returns the RMS magnitude error of the pilots over all data symbols.

Not available for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:CPE?`

Arguments None

Returns <NRf> data in percent.

FETCh:WLAN:SUMMary:EVM:PEAK:ALL? (Query Only)

Returns the peak EVM value for all subcarriers in the WLAN measurement or for all chips depending on the selected standard.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:EVM:PEAK:ALL?`

Arguments None

Returns <NRf> data in dB.

FETCh:WLAN:SUMMary:EVM:PEAK:CHIP:ALL? (Query Only)

Returns the chip at which the reported peak EVM value occurred.

Available only for 802.11b

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:EVM:PEAK:CHIP:ALL?`

Arguments None

Returns <NR1>

FETCh:WLAN:SUMMary:EVM:PEAK:CHIP:ONEKchips? (Query Only)

Returns the chip at which the reported peak EVM value over the first 1000 chips occurred.

Only available for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:EVM:PEAK:CHIP:ONEKchips?`

Arguments None

Returns <NR1>

FETCh:WLAN:SUMMary:EVM:PEAK:DATA? (Query Only)

Returns the peak EVM value for the data subcarriers in the WLAN measurement.

Not available for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:EVM:PEAK:DATA?`

Arguments None

Returns <NRf> data in dB.

FETCh:WLAN:SUMMary:EVM:PEAK:ONEKchips? (Query Only)

Returns the peak EVM value measured over the first 1000 chips.

Available only for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:EVM:PEAK:ONEKchips?`

Arguments None

Returns <NRf> data in dB.

FETCh:WLAN:SUMMary:EVM:PEAK:PILots? (Query Only)

Returns the peak EVM value for the pilots in the WLAN measurement.

Not available for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:EVM:PEAK:PILots?`

Arguments None

Returns <NRf> data in dB.

FETCh:WLAN:SUMMary:EVM:PEAK:SCARrier:ALL? (Query Only)

Returns the subcarrier on which the ALL peak EVM value occurred.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:EVM:PEAK:SCARrier:ALL?`

Arguments None

Returns <NR1>

FETCh:WLAN:SUMMary:EVM:PEAK:SCARrier:DATA? (Query Only)

Returns the subcarrier on which the DATA peak EVM value occurred.

Not available for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:EVM:PEAK:SCARrier:DATA?`

Arguments None

Returns <NR1>

FETCh:WLAN:SUMMary:EVM:PEAK:SCARrier:PILots? (Query Only)

Returns the subcarrier on which the PILOT peak EVM value occurred.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:EVM:PEAK:SCARrier:PILots?`

Arguments None

Returns <NR1>

FETCh:WLAN:SUMMary:EVM:PEAK:SYMBol:ALL? (Query Only)

Returns the symbol on which the reported EVM value occurred.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:EVM:PEAK:SYMBol:ALL?`

Arguments None

Returns <NR1>

FETCh:WLAN:SUMMary:EVM:PEAK:SYMBol:DATA? (Query Only)

Returns the symbol on which the DATA peak EVM value occurred.

Not available for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:EVM:PEAK:SYMBol:DATA?`

Arguments None

Returns <NR1>

FETCh:WLAN:SUMMary:EVM:PEAK:SYMBol:PILots? (Query Only)

Returns the symbol on which the PILOT peak EVM value occurred.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:EVM:PEAK:SYMBol:PILots?`

Arguments None

Returns <NR1>

FETCh:WLAN:SUMMary:EVM:RMS:ALL? (Query Only)

Returns the RMS EVM data for all subcarriers in the WLAN measurement and for all chips for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:EVM:RMS:ALL?`

Arguments None

Returns <NRf> RMS EVM in dB

FETCh:WLAN:SUMMary:EVM:RMS:AVERage? (Query Only)

Returns the average RMS EVM value in the WLAN measurement over the selected number of bursts.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:EVM:RMS:AVERage?`

Arguments None

Returns <NRf> RMS EVM in dB

FETCh:WLAN:SUMMary:EVM:RMS:DATA? (Query Only)

Returns the RMS EVM data for the data subcarriers in the WLAN measurement.

Not available for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:EVM:RMS:DATA?`

Arguments None

Returns <NRf> RMS EVM in dB

FETCh:WLAN:SUMMary:EVM:RMS:MAXimum? (Query Only)

Returns the maximum RMS EVMvalue in the WLAN measurement over the selected number of bursts.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:EVM:RMS:MAXimum?`

Arguments None

Returns <NRf> RMS EVM in dB

FETCh:WLAN:SUMMary:EVM:RMS:ONEKchips? (Query Only)

Returns the EVM RMSdata for the 1k chips in the WLAN measurement measured over the first 1000 chips.

Only available for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:EVM:RMS:ONEKchips?`

Arguments None

Returns <NRf> RMS EVM in dB

FETCh:WLAN:SUMMary:EVM:RMS:PILots? (Query Only)

Returns the RMS EVM data for the pilot subcarriers in the WLAN measurement.

Not available for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:EVM:RMS:PILots?`

Arguments None

Returns <NRf> data in percent.

FETCh:WLAN:SUMMary:FERRor? (Query Only)

Returns the frequency difference between the measured carrier frequency of the signal and the measurement frequency setting.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:FERRor?`

Arguments None

Returns <NRf> data in kHz.

FETCh:WLAN:SUMMary:HEADer? (Query Only)

Returns the received packet header information for the WLAN measurement. Applies to the 802.11b standard only.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:HEADer?`

Arguments None

Returns Results are returned in the same left-to-right order in which they are presented in the display.

FETCh:WLAN:SUMMary:HTSig? (Query Only)

Returns the decimal values of the received packet HT-SIGNAL. Applies to the 802.11n standard only.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:HTSig?`

Arguments None

Returns Results are returned in the same left-to-right order in which they are presented in the display.

FETCh:WLAN:SUMMary:IQOFfset? (Query Only)

Returns the average magnitude of the DC subcarrier level relative to total signal power.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:IQOFfset?`

Arguments None

Returns <NRf> data in dB.

FETCh:WLAN:SUMMary:LSIG? (Query Only)

Returns the decimal values of the received packet (legacy) SIGNAL symbols fields. Applies to the 802.11n standard only.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:LSIG?`

Arguments None

Returns Results are returned in the same left-to-right order in which they are presented in the display.

FETCh:WLAN:SUMMary:PACKet? (Query Only)

Returns the type and number of symbols, EVM-RMS and average power of the Preamble and Data portions of the packet.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:PACKet?`

Arguments None

Returns Results are returned in the same left-to-right order in which they are presented in the display.

FETCh:WLAN:SUMMary:PACKet:DMODulation? (Query Only)

Returns the modulation used in the Data symbols.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:PACKet:DMODulation?`

Returns <string> The data modulation being used.

FETCh:WLAN:SUMMary:PACKet:FORMat? (Query Only)

Returns the packet format setting for the WLAN measurement

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:PACKet:FORMat?`

Arguments None

Returns <string> The packet format setting.

FETCh:WLAN:SUMMary:PACKet:GUARd? (Query Only)

Returns the packet guard interval used by the DATA symbols.

Not available for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:PACKet:GUARd?`

Arguments None

Returns <NRf> The packet guard interval, 1/4 or 1/8.

FETCh:WLAN:SUMMary:SIG? (Query Only)

Returns the decimal values of the received packet SIGNAL symbols fields.

Available only for 802.11agjp.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:SIG?`

Arguments None

Returns Results are returned in the same left-to-right order in which they are presented in the display.

FETCh:WLAN:SUMMary:VHTSig:A? (Query Only)

Returns the decimal values of the received packet VHT-SIGNAL(A) symbols fields.

Available only for 802.11ac.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:VHTSig:A?`

Arguments None

Returns Results are returned in the same left-to-right order in which they are presented in the display.

FETCh:WLAN:SUMMary:VHTSig:B? (Query Only)

Returns the decimal values of the received packet VHT-SIGNAL(B) symbols fields.

Available only for 802.11ac.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Fetch commands

Syntax `FETCh:WLAN:SUMMary:VHTSig:B?`

Arguments None

Returns Results are returned in the same left-to-right order in which they are presented in the display.

*IDN? (Query Only)

Returns the analyzer identification code.

Conditions Measurement views: All

Group IEEE common commands

Syntax `*IDN?`

Arguments None

Returns The analyzer identification code in the following format

```
TEKTRONIX,RSAXXXXX,<serial_number>,<firmware_version>
```

Where
`TEKTRONIX` indicates that the manufacturer is Tektronix.
`RSAXXXXX` is the model number (RSA5103B, RSA5106B, RSA5115B, or RSA5126B).
`<serial_number>` is the serial number.
`<firmware_version>` is the firmware version.

Examples `*IDN?` might return the response
`TEKTRONIX,RSA5126B,B000111,FV2.6.1500`.

INITiate:CONTinous (No Query Form)

Determines whether to use the continuous mode to acquire the input signal. You must then use the `INITiate[:IMMediate]` command to start play.

Group Initiate commands

Syntax `INITiate:CONTinous { 1 | 0 | ON | OFF }`

Related Commands INITiate[:IMMediate]

Arguments 1 sets play to ON.

0 sets play to OFF.

INITiate:CONTinuous

Determines whether to place the analyzer in the single or the continuous acquisition mode.

Conditions Measurement views: All

Group Initiate commands

Syntax

```
INITiate:CONTinuous { OFF | ON | 0 | 1 }
INITiate:CONTinuous?
```

Related Commands INITiate:RESumeINITiate[:IMMediate]

Arguments `OFF` or `0` places the analyzer in the single acquisition mode. To initiate the acquisition, use the INITiate[:IMMediate] command. To stop acquisition when a trigger is not present, and to avoid a 2.7 hour time-out, send the free-run trigger command, TRIGger:SEQuence:STATus 0. Then, send the command INITiate:CONTinuous OFF. This sequence ensures a minimal delay before the INIT:CONT OFF command completes.

`ON` or `1` places the analyzer in the continuous acquisition mode. To initiate the acquisition, use the INITiate[:IMMediate] command. To stop the acquisition in the continuous mode, send the following command: INITiate:CONTinuous OFF

Examples `INITIATE:CONTINUOUS ON` places the analyzer in the continuous acquisition mode.

INITiate[:IMMediate] (No Query Form)

Starts input signal acquisition.

NOTE. *It is an overlapped command, which does not finish executing before the next command starts executing. Use the *OPC(?) and *WAI commands to synchronize all pending operations to the execution of this command.*

Conditions Measurement views: All

Group Initiate commands

Syntax `INITiate[:IMMediate]`

Related Commands *OPC, *TRG, *WAI, INITiate:CONTinuous

Arguments None

Examples `INITIATE:IMMEDIATE` starts input signal acquisition.

INITiate:RESume (No Query Form)

Restarts signal processing.

Resume is only applicable when FastFrame is enabled, and when the instrument is stopped while analyzing the frames. When the instrument is stopped, measurements will finish processing and the system might stop before all of the frames of an acquisition are processed (when acquisition is stopped, the system stops as soon as all measurements say they are finished processing the current acquisition). However, the instrument saves all of the frames to the Acquisition History (Data Cache), and when INITiate:RESume is sent, it will begin processing at the first frame of the last acquisition that was not processed (the one after the last frame processed), and process all of these remaining frames before starting a new acquisition. If INITiate is sent instead, any remaining frames are discarded and a new acquisition is started immediately. If FastFrame is off, or there are no remainder frames, or the acquisition was aborted rather than stopped, then INITiate:RESume and INITiate are the same.

Conditions Measurement views: All

FastFrame enabled

Group Initiate commands

Syntax `INITiate:RESume`

Related Commands *OPC, *TRG, *WAI, INITiate:CONTinuousINITiate[:IMMediate]

Arguments None

Examples `INITIATE:RESUME` restarts input signal acquisition.

INPut:CORRection:EXTernal:EDIT<x>:INTerpolation

Sets or queries the interpolation setting to use with the indicated external gain table.

Conditions Measurement views: All

Group Input commands

Syntax
```
INPut:CORRection:EXTernal:EDIT<x>:INTerpolation { LINear | LOGarithmic }
INPut:CORRection:EXTernal:EDIT<x>:INTerpolation
```

Related Commands INPut:CORRection:EXTernal:EDIT<x>:NEW

Arguments `LINear` select when the frequency scale of the spectrum or spurious measurements are linear.

`LOGarithmic` select when the frequency scale of the spectrum is logarithmic.

Examples `INPut:CORRection:EXTernal:EDIT1:INTerpolation LINear` uses linear interpolation when applying External correction Table one (1) to the chosen waveforms.

INPut:CORRection:EXTernal:EDIT<x>:NEW

Creates the indicated external loss table from a series of frequency and gain value pairs.

Conditions Measurement views: All

Group Input commands

Syntax

```
INPut:CORRection:EXTernal:EDIT<x>:NEW
<freq(1)>,<gain(1)>,<freq(2)>,<gain(2)>,...,<freq(n)>,<gain(n)>
INPut:CORRection:EXTernal:EDIT<x>:NEW?
```

Related Commands INPut:CORRection:EXTernal:TYPE

Arguments `<freq(n)>,<gain(n)>` specifies a pair of frequency (<NR3> in Hz) and gain (+ | -<NR3> in dB) in the external correction table.

NOTE. *Do not include a unit suffix (such as Hz, dBm, or dB) with the arguments for this command.*

The setting range is:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Loss: -50 to +30 dB. Negative values indicate loss and positive values indicate gain. Resolution is 0.1 dB.

Examples `INPut:CORRection:EXTernal:EDIT?:NEW 1.0E+9,-2.2, -1.5E+9,-2.3` creates the External correction Table specifying a loss of 2.2 dB at 1 GHz and a loss of 2.3 dB at 1.5 GHz.

INPut:CORRection:EXTernal:TYPE

Sets or queries the data type to use when applying the external loss table corrections.

Conditions Measurement views: All

Group Input commands

Syntax

```
INPut:CORRection:EXTernal:TYPE { TRACe | DATA }
INPut:CORRection:EXTernal:TYPE
```

Related Commands INPut:CORRection:EXTernal:EDIT<x>:NEW

Arguments

TRACe selects traces in the Spectrum, Spectrogram, Spurious, and Amplitude versus Time views. Selecting TRACe disables the CALibration:CORRection:EXTernal:GAIN:STATe command.

DATA selects all acquired data. Selecting DATA enables the CALibration:CORRection:EXTernal:GAIN:STATe command.

Examples

INPUT:CORRECTION:EXTERNAL:TYPETRACe selects traces to apply the external loss table corrections.

INPut:{MLEVel|RLEVel}

Sets or queries the reference level.

Conditions

Measurement views: All

Group

Input commands

Syntax

```
INPut:{MLEVel|RLEVel} <value>
INPut:{MLEVel|RLEVel}?
```

Arguments

<value> :: <NRf> specifies the reference level. Range: -170 to +50 dBm.

Examples

INPUT:RLEVEL -10 sets the reference level to -10 dBm.

INPut[:RF]:ATTenuation

Sets or queries the input attenuation. Programming a specified attenuation sets INPut[:RF]:ATTenuation:AUTO OFF.

Conditions

Measurement views: All

Group

Input commands

Syntax

```
INPut[:RF]:ATTenuation <value>
INPut[:RF]:ATTenuation?
```

Related Commands

INPut[:RF]:ATTenuation:AUTO

Arguments `<value> :: <NR1>` specifies the input attenuation.
Range: 0 to 75 dB in 5 dB steps.

Examples `INPUT:RF:ATTENUATION 20` sets the input attenuation to 20 dB.

INPut[:RF]:ATTenuation:AUTO

Determines whether to set the input attenuation automatically or manually.

Conditions Measurement views: All

Group Input commands

Syntax
```
INPut[:RF]:ATTenuation:AUTO { OFF | ON | 0 | 1 }
INPut[:RF]:ATTenuation:AUTO?
```

Arguments `OFF` or `0` specifies that the input attenuation is set manually.
To set it, use the INPut[:RF]:ATTenuation command.

`ON` or `1` specifies that the input attenuation is set automatically according to the reference level.

Examples `INPUT:RF:ATTENUATION:AUTO ON` specifies that the input attenuation is set automatically.

INPut[:RF]:ATTenuation:MONitor:STATe

Determines whether to enable or disable to monitor attenuator use.

Conditions Measurement views: All

Group Input commands

Syntax
```
INPut[:RF]:ATTenuation:MONitor:STATe { OFF | ON | 0 | 1 }
INPut[:RF]:ATTenuation:MONitor:STATe?
```

Arguments `OFF` or `0` disables to monitor attenuator use.

`ON` or `1` enables to monitor attenuator use.

Examples `INPUT:RF:ATTENUATION:MONITOR:STATE ON` enables to monitor attenuator use.

INPut[:RF]:GAIN:STATe

Determines whether to enable or disable the internal preamp.

Conditions Measurement views: All (Option 50 or 51 only)

Group Input commands

Syntax
```
INPut[:RF]:GAIN:STATe { OFF | ON | 0 | 1 }
INPut[:RF]:GAIN:STATe?
```

Arguments `OFF` or `0` disables the internal preamp.

`ON` or `1` enables the internal preamp.

Examples `INPUT:RF:GAIN:STATE ON` enables the internal preamp.

MMEMory:{AM|FM|PM}:LOAD:TRACe (No Query Form)

Loads the AM/FM/PM measurement trace data from the specified file.

Conditions Measurement views: AM, FM, PM

Group Mass memory commands

Syntax `MMEMory:{AM|FM|PM}:LOAD:TRACe <file_name>`

Arguments `<file_name> :: <string>` specifies the file from which to load trace data. The file extension is .AmVsTime (AM), .FmVsTime (FM), or .PmVsTime (PM). You can omit the extension.

Examples `MMEMORY:AM:LOAD:TRACE "Sample1"` loads the trace data from the *Sample1.AmVsTime* file in the AM measurement.

MMEMory:{AM|FM|PM}:SHOW:TRACe<x>

Enables display of a recalled trace file in Trace<x>. The result is the same as selecting “Show recalled trace” in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions Measurement views: AM, FM, PM

Group Mass memory commands

Syntax

```
MMEMory:{AM|FM|PM}:SHOW:TRACe<x> { OFF | ON | 0 | 1 }
MMEMory:{AM|FM|PM}:SHOW:TRACe<x>?
```

Related Commands MMEMory:{AM|FM|PM}:LOAD:TRACe

Arguments `OFF` or `0` disables display of the recalled trace.

`ON` or `1` enables display of the recalled trace.

Examples `MMEMORY:FM:SHOW:TRACE2 ON` enables display of the recalled data loaded in Trace2.

MMEMory:{AM|FM|PM}:STORe:TRACe (No Query Form)

Stores the AM/FM/PM measurement trace data in the specified file.

Conditions Measurement views: AM, FM, PM

Group Mass memory commands

Syntax

```
MMEMory:{AM|FM|PM}:STORe:TRACe <file_name>
```

Arguments `<file_name> :: <string>` specifies the file from which to load trace data. The file extension is .AmVsTime (AM), .FmVsTime (FM), or .PmVsTime (PM). You can omit the extension.

Examples `MMEMORY:AM:STORE:TRACE "Sample1"` stores the trace data in the *Sample1.AmVsTime* file in the AM measurement.

MMEMory:AUDio:LOAD:FILTer (No Query Form)

Loads a user-defined audio filter from the specified file.

Conditions Measurement views: Audio Spectrum

Group Mass memory commands

Syntax `MMEMory:AUDio:LOAD:FILTer <file_name>`

Related Commands [SENSe]:AUDio:FILTer:SOURce?

Arguments <file_name> ::= <string> specifies the filter file to load.

Examples `MMEMORY:AUDIO:LOAD:FILTER` "filter4" loads the user-defined filter4 file for the measurement.

MMEMory:AVTime:LOAD:TRACe<x> (No Query Form)

Loads the Amplitude versus Time trace data from the specified file.

The parameter <x> = 1 to 3; Trace 4 (math trace) is invalid.

Conditions Measurement views: Amplitude versus Time

Group Mass memory commands

Syntax `MMEMory:AVTime:LOAD:TRACe<x> <file_name>`

Arguments `<file_name> :: <string>` specifies the file from which to load trace data. The file extension is .AmplVsTime. You can omit the extension.

Examples `MMEMORY:AVTIME:LOAD:TRACE1 "Sample1"` loads Trace 1 data from the *Sample1.AmplVsTime* file.

MMEMory:AVTime:SHOW:TRACe<x>

Enables display of a recalled trace file in Trace<x>. The result is the same as selecting "Show recalled trace" in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions Measurement views: Amplitude versus Time

Group Mass memory commands

Syntax
```
MMEMory:AVTime:SHOW:TRACe<x> { OFF | ON | 0 | 1 }
MMEMory:AVTime:SHOW:TRACe<x>?
```

Related Commands MMEMory:AVTime:LOAD:TRACe<x>

Arguments `OFF` or `0` disables display of the recalled trace.

`ON` or `1` enables display of the recalled trace.

Examples `MMEMORY:AVTIME:SHOW:TRACE2 ON` enables display of the recalled data loaded in Trace2.

MMEMory:AVTime:STORe:TRACe<x> (No Query Form)

Stores the Amplitude versus Time trace data in the specified file.

The parameter <x> = 1 to 4; All traces are valid.

Conditions Measurement views: Amplitude versus Time

Group Mass memory commands

Syntax
```
MMEMory:AVTime:STORe:TRACe<x> <file_name>
```

Arguments `<file_name> :: <string>` specifies the file in which to store trace data. The file extension is .AmplVsTime. You can omit the extension.

Examples `MMEMORY:AVTIME:STORE:TRACE1 "Sample1"` stores Trace 1 data in the *Sample1.AmplVsTime* file.

MMEMory:CALibration:LOAD:CORRection:EXTernal:EDIT<x> (No Query Form)

Loads the external loss table from a specified file.

The parameter <x> = 1 to 3 represent External Loss Table 1 to 3, respectively.

Conditions Measurement views: All

Group Mass memory commands

Syntax
```
MMEMory:CALibration:LOAD:CORRection:EXTernal:EDIT<x>
<file_name>
```

Arguments `<file_name> :: <string>` specifies the file to load the external loss table from.
The file extension is .csv. You can omit the extension.

Examples
```
MMEMORY:CALIBRATION:LOAD:CORRECTION:EXTERNAL:EDIT1 "Table1"
```
loads the External Loss Table 1 from the *Table1.csv* file.

MMEMory:CALibration:STORe:CORRection:EXTernal:EDIT<x> (No Query Form)

Stores the external loss table to a specified file.

The parameter <x> = 1 to 3 represent External Loss Table 1 to 3, respectively.

Conditions Measurement views: All

Group Mass memory commands

Syntax
```
MMEMory:CALibration:STORe:CORRection:EXTernal:EDIT<x>
<file_name>
```

Arguments `<file_name> :: <string>` specifies the file to store the external loss table to.
The file extension is .csv. You can omit the extension.

Examples
```
MMEMORY:CALIBRATION:STORE:CORRECTION:EXTERNAL:EDIT1 "Table1"
```
stores the External Loss Table 1 to the *Table1.csv* file.

MMEMory:CCDF:LOAD:TRACe<x> (No Query Form)

Loads the CCDF trace data from the specified file.

Conditions Measurement views: CCDF

Group Mass memory commands

Syntax `MMEMory:CCDF:LOAD:TRACe<x> <file_name>`

Arguments `<file_name> :: <string>` specifies the file from which to load trace data. The file extension is .CCDF. You can omit the extension.

Examples `MMEMORY:CCDF:LOAD:TRACE1 "Sample1"` loads Trace 1 data from the *Sample1.CCDF* file.

MMEMory:CCDF:SHOW:TRACe<x>

Enables display of a recalled trace file in Trace<x>. The result is the same as selecting “Show recalled trace” in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions Measurement views: CCDF

Group Mass memory commands

Syntax
```
MMEMory:CCDF:SHOW:TRACe<x> { OFF | ON | 0 | 1 }
MMEMory:CCDF:SHOW:TRACe<x>?
```

Related Commands MMEMory:CCDF:LOAD:TRACe<x>

Arguments `OFF` or `0` disables display of the recalled trace.

`ON` or `1` enables display of the recalled trace.

Examples `MMEMORY:CCDF:SHOW:TRACE2 ON` enables display of the recalled data loaded in Trace2.

MMEMory:CCDF:STORe:TRACe<x> (No Query Form)

Stores the CCDF trace data in the specified file.

Conditions Measurement views: CCDF

Group Mass memory commands

Syntax `MMEMory:CCDF:STORe:TRACe<x> <file_name>`

Arguments `<file_name> :: <string>` specifies the file in which to store trace data. The file extension is .CCDF. You can omit the extension.

Examples `MMEMORY:CCDF:STORE:TRACE1 "Sample1"` stores Trace 1 data in the *Sample1.CCDF* file.

MMEMory:DDEMod:LOAD:FILTer:MEASurement:UOTHer (No Query Form)

Loads a user-defined measurement filter from the specified file.

Group Mass memory commands

Syntax `MMEMory:DDEMod:LOAD:FILTer:MEASurement:UOTHer <file_name>`

Related Commands [SENSe]:DDEMod:FILTer:REFerence

[SENSe]:DDEMod:FILTer:MEASurement

Arguments `<file_name> :: <string>`specifies the filter file to load.

Examples `MMEMORY:DDEMOD:LOAD:FILTER:MEASUREMENT:UOTHER "Table2"` loads User other filter from the file Table2.csv.

MMEMory:DDEMod:LOAD:FILTer:MEASurement:USER<x> (No Query Form)

Loads the user-defined measurement filter from the specified file. The parameter <x> specifies one of the User Meas Filters, 1 to 3. The filter file must have a .csv file extension, though you may omit the file extension in the command line.

Group Mass memory commands

Syntax `MMEMory:DDEMod:LOAD:FILTer:MEASurement:USER<x> <file_name>`

Related Commands [SENSe]:DDEMod:FILTer:REFerence

[SENSe]:DDEMod:FILTer:MEASurement

Arguments `<file_name> :: <string>` specifies the filter file to load.

Examples `MMEMORY:DDEMOD:LOAD:FILTER:MEASUREMENT:USER1 "Table1"` loads User Meas Filter 1 from the file Table1.csv.

MMEMory:DDEMod:LOAD:FILTer:REFerence:UOTHer (No Query Form)

Loads the specified user-defined reference filter, UOTHer, from a specified file.

Group Mass memory commands

Syntax `MMEMory:DDEMod:LOAD:FILTer:REFerence:UOTHer <file_name>`

Related Commands [SENSe]:DDEMod:FILTer:REFerence

[SENSe]:DDEMod:FILTer:MEASurement

Arguments `<file_name> :: <string>` specifies the reference filter file to load. The file extension is type .csv.

Examples `MMEMORY:DDEMOD:LOAD:FILTER:REFERENCE:UOTHER "FinalFilt"` loads the filter file FinalFilt.csv into the filter reference UOTHER.

MMEMory:DDEMod:LOAD:FILTer:REFerence:USER<x> (No Query Form)

Loads the specified user-defined reference filter, USER1 | 2 | 3 from a specified file.

Group Mass memory commands

Syntax `MMEMory:DDEMod:LOAD:FILTer:REFerence:USER<x> <file_name>`

Related Commands [SENSe]:DDEMod:FILTer:REFerence

Arguments <file_name> :: <string>specifies the reference filter file to load. The file extension is type .csv.

Examples MMEMORY:DDEMOD:LOAD:FILTER:REFERENCE:USER2 "DUTFilt2" loads the filter file DUTfilt2.csv into the filter reference USER2.

MMEMory:DDEMod:LOAD:SYMBol:MAP (No Query Form)

Loads the specified symbol map filename. The setting correlates to the field, User Symbol Map (per Modulation Type) on the Advanced Params tab in the Digital Modulation settings.

Group Mass memory commands

Syntax MMEMory:DDEMod:LOAD:SYMBol:MAP <file_name>

Related Commands [SENSe]:DDEMod:MODulation:TYPE

Arguments <file_name> :: <string>specifies the symbol map file to load. The file extension is type .txt.

Examples MMEMORY:DDEMOD:LOAD:SYMBOL:MAP "C:\usymmap.txt" identifies the file to load for the symbol map.

MMEMory:DPX:LOAD:TRACe<x> (No Query Form)

Loads the DPX spectrum trace data from the specified file.

The parameter <x> = 1, 2, 3, or 5; TRACe4 (math trace) is invalid.

Conditions Measurement views: DPX spectrum

Group Mass memory commands

Syntax MMEMory:DPX:LOAD:TRACe<x> <file_name>

Arguments `<file_name> :: <string>` specifies the file from which to load trace data. The file extension is .dpt. You can omit the extension.

Examples `MMEMORY:DPX:LOAD:TRACE1 "Sample1"` loads Trace 1 data from the *Sample1.dpt* file.

MMEMory:DPX:SHOW:TRACe<x>

Enables display of a recalled trace file in Trace<x>. The result is the same as selecting "Show recalled trace" in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions Measurement views: DPX spectrum

Group Mass memory commands

Syntax
```
MMEMory:DPX:SHOW:TRACe<x> { OFF | ON | 0 | 1 }
MMEMory:DPX:SHOW:TRACe<x>?
```

Related Commands MMEMory:DPX:LOAD:TRACe<x>

Arguments `OFF` or `0` disables display of the recalled trace.

`ON` or `1` enables display of the recalled trace.

Examples `MMEMORY:DPX:SHOW:TRACE2 ON` enables display of the recalled data loaded in Trace2.

MMEMory:DPX:STORe:TRACe<x>

Stores the DPX spectrum trace data in the specified file.

The parameter <x> = 1 to 5; All traces are valid.

Conditions Measurement views: DPX spectrum

Group Mass memory commands

Syntax `MMEMory:DPX:STORe:TRACe<x> <file_name>`

Arguments `<file_name> :: <string>` specifies the file in which to store trace data. The file extension is .dpt. You can omit the extension.

Examples `MMEMORY:DPX:STORE:TRACE1 "Sample1"` stores Trace 1 data in the *Sample1.dpt* file.

MMEMory:DPX:TDM:FREQuency:TXBer:LOAD:PATTern (No Query Form)

Loads the Tx BER User data pattern from the specified file.

Conditions Measurement views: DPX:Frequency

Group Mass memory commands

Syntax `MMEMory:DPX:TDM:FREQuency:TXBer:LOAD:PATTern <file_name>`

Related Commands [SENSe]:DPX:TDM:FREQuency:TXBer:PATTern, [SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:REPeat

Arguments `<file_name> :: <string>` that specifies the file (path) from which to load the User data pattern.
The file type containing the data pattern must be in CSV format. See the instrument Help for details on creating a user pattern file.

Examples `MMEMORY:DPX:TDM:FREQUENCY:TXBER:LOAD:PATTERN "C:\RSA5100B Files\Example Files\SampleUserPattern.csv"` loads the user pattern defined in SampleUserPattern.csv.

[MMEMory:{FSETtling|PSETtling}:LOAD:TRACe<x>

Loads the Frequency or Phase Settling Time trace data from the specified file. The parameter <x> = 1 or 2; only Trace1 and Trace2 are used for settling time measurements.

Conditions Measurement views: Frequency or Phase Settling

Group Mass memory commands

Syntax `[MMEMory:{FSETtling|PSETtling}:LOAD:TRACe<x> <file_name>`

Arguments `<file_name> :: <string>` specifies the file from which to load trace data.

Examples `MMEMORY:PSETTLING:LOAD:TRACE1 "Wfm1"` loads Trace1 data from the Wfm1 file.

[MMEMory:{FSETtling|PSETtling}:SHOW:TRACe<x>

Enables display of a recalled Frequency or Phase Settling trace file in Trace<x>. The result is the same as selecting "Show recalled trace" in the Trace control panel.

Conditions Measurement views: Frequency and Phase Settling

Group Mass memory commands

Syntax
```
[MMEMory:{FSETtling|PSETtling}:SHOW:TRACe<x> { OFF | ON | 0 | 1 }
[MMEMory:{FSETtling|PSETtling}:SHOW:TRACe<x>?
```

Related Commands [MMEMory:{FSETtling|PSETtling}:LOAD:TRACe<x>

Arguments `OFF` or `0` disables display of the recalled trace.

`ON` or `1` enables display of the recalled trace.

Examples `MMEMORY:PSETTLING:SHOW:TRACE1 ON` enables display of the recalled data loaded in Trace1.

[MMEMory:{FSETtling|PSETtling}:STORe:TRACe<x>

Stores the Frequency or Phase Settling trace data in the specified file. The parameter <x> = 1 or 2; only Trace1 and Trace2 are used for Frequency and Phase Settling.

Conditions Measurement views: Phase noise

Group Mass memory commands

Syntax `[MMEMory:{FSETtling|PSETtling}:STORe:TRACe<x> <file_name>`

Arguments `<file_name> ::  <string>` specifies the file in which to store trace data.

Examples `MMEMORY:PNOISE:STORE:TRACE1 "TRACE1"` stores Trace 1 data in the TRACE1 file.

MMEMory:FVTime:LOAD:TRACe (No Query Form)

Loads the Frequency versus Time trace data from the specified file.

Conditions Measurement views: Frequency versus Time

Group Mass memory commands

Syntax `MMEMory:FVTime:LOAD:TRACe <file_name>`

Arguments `<file_name> ::  <string>` specifies the file from which to load trace data. The file extension is .FreqVsTime. You can omit the extension.

Examples `MMEMORY:FVTIME:LOAD:TRACE"Sample1"` loads the Frequency versus Time trace data from the *Sample1.FreqVsTime* file.

MMEMory:FVTIME:SHOW:TRACe<x>

Enables display of a recalled trace file in Trace<x>. The result is the same as selecting “Show recalled trace” in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions Measurement views: Frequency versus Time

Group Mass memory commands

Syntax
```
MMEMory:FVTIME:SHOW:TRACe<x> { OFF | ON | 0 | 1 }
MMEMory:FVTIME:SHOW:TRACe<x>?
```

Related Commands MMEMory:FVTime:LOAD:TRACe

Arguments OFF or 0 disables display of the recalled trace.

ON or 1 enables display of the recalled trace.

Examples MMEMORY:FVTIME:SHOW:TRACE2 ON enables display of the recalled data loaded in Trace2.

MMEMory:FVTime:STORe:TRACe (No Query Form)

Stores the Frequency versus Time trace data in the specified file.

Conditions Measurement views: Frequency versus Time

Group Mass memory commands

Syntax MMEMory:FVTime:STORe:TRACe <file_name>

Arguments <file_name> :: <string> specifies the file in which to store trace data. The file extension is .FreqVsTime. You can omit the extension.

Examples MMEMORY:FVTIME:STORE:TRACE"Sample1" stores the Frequency versus Time trace data in the *Sample1.FreqVsTime* file.

MMEMory:IQVTime:LOAD:TRACe:I (No Query Form)

Loads I trace data from the specified file.

Conditions Measurement views: RF I&Q versus Time

Group Mass memory commands

Syntax MMEMory:IQVTime:LOAD:TRACe:I <file_name>

Arguments <file_name> :: <string> specifies the file from which to load trace data. The file extension is .RFIQVsTime. You can omit the extension.

Examples `MMEMORY:IQVTIME:LOAD:TRACE:I"Sample1"` loads the I trace data from the *Sample1.RFIQVsTime* file.

MMEMory:IQVTime:LOAD:TRACe:Q (No Query Form)

Loads Q trace data from the specified file.

Conditions Measurement views: RF I&Q versus Time

Group Mass memory commands

Syntax `MMEMory:IQVTime:LOAD:TRACe:Q <file_name>`

Arguments `<file_name> :: <string>` specifies the file from which to load trace data. The file extension is .RFIQVsTime. You can omit the extension.

Examples `MMEMORY:IQVTIME:LOAD:TRACE:Q"Sample2"` loads the I trace data from the *Sample2.RFIQVsTime* file.

MMEMory:IQVTIME:SHOW:TRACe:I

Enables display of a recalled trace file in Trace. The result is the same as selecting "Show recalled trace" in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions Measurement views: RF I&Q versus Time

Group Mass memory commands

Syntax
```
MMEMory:IQVTIME:SHOW:TRACe:I { OFF | ON | 0 | 1 }
MMEMory:IQVTIME:SHOW:TRACe:I?
```

Related Commands MMEMory:IQVTime:LOAD:TRACe:I

Arguments `OFF` or `0` disables display of the recalled trace.

`ON` or `1` enables display of the recalled trace.

Examples `MMEMORY:IQVTIME:SHOW:TRACE:I ON` enables display of the recalled data loaded in the Trace.

MMEMory:IQVTIME:SHOW:TRACe<x>:Q

Enables display of a recalled trace file in Trace<x>. The result is the same as selecting "Show recalled trace" in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions Measurement views: RF I&Q versus Time

Group Mass memory commands

Syntax
```
MMEMory:IQVTIME:SHOW:TRACe<x>:Q { OFF | ON | 0 | 1 }
MMEMory:IQVTIME:SHOW:TRACe<x>:Q?
```

Related Commands MMEMory:IQVTIME:SHOW:TRACe<x>:Q

Arguments `OFF` or `0` disables display of the recalled trace.

`ON` or `1` enables display of the recalled trace.

Examples `MMEMORY:IQVTIME:SHOW:TRACE3:Q ON` enables display of the recalled data loaded in Trace3.

MMEMory:IQVTime:STORe:TRACe:I (No Query Form)

Stores I trace data in the specified file.

Conditions Measurement views: RF I&Q versus Time

Group Mass memory commands

Syntax
```
MMEMory:IQVTime:STORe:TRACe:I <file_name>
```

Arguments `<file_name> :: <string>` specifies the file in which to store trace data. The file extension is .RFIQVsTime. You can omit the extension.

Examples `MMEMORY:IQVTIME:STORE:TRACE:I"Sample1"` stores the I trace data in the *Sample1.RFIQVsTime* file.

MMEMory:IQVTime:STORe:TRACe:Q (No Query Form)

Stores Q trace data in the specified file.

Conditions Measurement views: RF I&Q versus Time

Group Mass memory commands

Syntax `MMEMory:IQVTime:STORe:TRACe:Q <file_name>`

Arguments `<file_name> :: <string>` specifies the file in which to store trace data. The file extension is .RFIQVsTime. You can omit the extension.

Examples `MMEMORY:IQVTIME:STORE:TRACE:Q"Sample2"` stores the Q trace data in the *Sample2.RFIQVsTime* file.

MMEMory:LOAD:IQ (No Query Form)

Loads time-domain IQ waveform into the acquisition memory from a file.

Conditions Measurement views: All

Group Mass memory commands

Syntax `MMEMory:LOAD:IQ <file_name>`

Arguments `<file_name> :: <string>` specifies the file to load IQ data from. The file extension is .tiq. You can omit the extension.

Examples `MMEMORY:LOAD:IQ"IQ1"` loads IQ data from the *IQ1.tiq* file.

MMEMory:LOAD:STATe (No Query Form)

Loads the instrument setup from a specified file for the currently selected view.

Conditions Measurement views: All

Group Mass memory commands

Syntax `MMEMory:LOAD:STATe <file_name>`

Arguments `<file_name> :: <string>` specifies the file to load the instrument setup from.
The file extension is .setup. You can omit the extension.

Examples `MMEMORY:LOAD:STATE"STATE1"` loads the instrument setup from the *STATE1.setup* file.

MMEMory:LOAD:TRACe (No Query Form)

Loads the trace data from a specified file for the currently selected view.

Conditions Measurement views (See Table 2-26.)

Group Mass memory commands

Syntax `MMEMory:LOAD:TRACe <file_name>`

Related Commands MMEMory:STORe:TRACe

Arguments `<file_name> :: <string>` specifies the file to load the trace data from. The file extension is named for the measurement view as shown in the following table. The trace file is not available in some views which are indicated by "NA" in the file extension column.

For the views that allow multiple-trace display, select the trace to load or store using the command listed in the trace selection column in the following table. You can load the specified trace with any trace data that you saved in the view if the traces have the same units. For example, you can load Trace 2 with Trace 1 data that you saved.

Table 2-26: Trace file extension and trace selection command

Display group	Measurement view	File extension	Trace selection
General signal viewing	Spectrum	Specan	TRACe<x>:SPECtrum:SELect
	DPX spectrum	dpt	TRACe<x>:DPX:SELect
	Amplitude versus Time	AmplVsTime	TRACe<x>:AVTime:SELect
	Frequency versus Time	FreqVsTime	NA
	Phase versus Time	PhaseVsTime	NA
	RF I&Q versus Time	RFIQVsTime	NA
	Spectrogram	Sogram	NA
	Time overview	NA	NA
General purpose analog modulation (Option 21 only)	AM	AmVsTime	NA
	FM	FmVsTime	NA
	PM	PmVsTime	NA
General purpose digital modulation (Option 21 only)	Constellation	NA	NA
	Demodulated I & Q versus Time	NA	NA
	Eye Diagram	NA	NA
	EVM versus Time	NA	NA
	Frequency Deviation versus Time	NA	NA
	Magnitude error versus Time	NA	NA
	Phase error versus Time	NA	NA
	Signal quality	NA	NA
	Symbol table	NA	NA
	Trellis Diagram	NA	NA
RF measurements	CCDF	CCDF	TRACe<x>:CCDF:SELect
	Channel power and ACPR	NA	NA
	MCPR	NA	NA
	Occupied Bandwidth	NA	NA
	Phase noise (Option 11 only)	PhaseNoise	TRACe<x>:{FSETtling|PSETtling}:SELect
	Spurious	NA	NA
Pulsed RF	Pulse statistics	NA	NA
	Pulse table	NA	NA
	Pulse trace	NA	NA

Examples `MMEMORY:LOAD:TRACE"TRACE1"` loads the trace data from the *TRACE1.Specan* file when the spectrum view is selected.

MMEMory:LTE:ACLR:LOAD:TABLe (No Query Form)

Loads the LTE ACLR table containing the limits for enabled ranges from the specified .cvs file.

Conditions Measurement view: LTE ACLR

Group Mass memory commands

Syntax `MMEMory:LTE:ACLR:LOAD:TABLe <string>`

Arguments <file_name> :: <string> specifies the file from which to load the LTE ACLR table. The file extension is .csv. You can omit the extension.

Examples `MMEMORY:LTE:ACLR:LOAD:TABLE` "Table1" loads the LTE ACLR table from the Table1.csv file.

MMEMory:LTE:ACLR:STORe:TABLe (No Query Form)

Stores the LTE ACLR table containing the limits for enabled ranges in a specified .cvs file, allowing you to export the file into Microsoft Excel or other database system.

Conditions Measurement view: LTE ACLR

Group Mass memory commands

Syntax `MMEMory:LTE:ACLR:STORe:TABLe <string>`

Arguments <file_name> :: <string> specifies the file in which to store the LTE ACLR table. The file extension is .csv. You can omit the extension.

Examples `MMEMORY:LTE:ACLR:STORE:TABLE` "Table1" stores the LTE ACLR table in the Table1.csv file.

MMEMory:MAPit:LOAD (No Query Form)

Loads the MapIt measurement results from the specified file.

Conditions Measurement views: Spectrum, DPX, Spectrogram, Amplitude vs Time, Channel Power, Occupied Bandwidth

Group Mass memory commands

Syntax `MMEMory:MAPit:LOAD <filename>`

Arguments <filename> :: <string> the name of the file from which to load the MapIt data.

Examples `MMEMORY:MAPIT:LOAD "C:\SignalVu-PC files\MapIt_results1"` Loads the MapIt measurement results in the MapIt_results1 file from the C:SignalVu-PC files folder.

MMEMory:MAPit:STORe (No Query Form)

Save the selected MapIt measurement results to a file. The results to save are selected by the <result_index>. The saved files are compressed .zip files.

Conditions Measurement views: Spectrum, DPX, Spectrogram, Amplitude vs Time, Channel Power, Occupied Bandwidth

Group Mass memory commands

Syntax `MMEMory:MAPit:STORe <filename>, <result_index> [, <result_index]`

Arguments <filenam> :: <string>the name of the file in which to store the MapIt measurement results.

<result_index> :: <NRf>. The 1-based index of the MapIt result to load. See the MapIt editor for the list of results. `index 1` is the first result at the top of the list.

Examples `MMEMORY:MAPIT:STORE` ??

MMEMory:MAPit:STORe:ALL (No Query Form)

Saves all of the MapIt measurement results from memory to the specified file.

Conditions Measurement views: Spectrum, DPX, Spectrogram, Amplitude vs Time, Channel Power, Occupied Bandwidth

Group Mass memory commands

Syntax `MMEMory:MAPit:STORe:ALL <filename>`

Arguments <filename> :: <string> the name of the file in which to store the MapIt measurement results.

Examples `MMEMORY:MAPIT:STORE:ALL "C:\SignalVu-PC files\MapIt_results1"` Stores the MapIt measurement results in the MapIt_results1 file in the C:SignalVu-PC files folder.

MMEMory:NOISe:GAIN:LOAD:TRACe (No Query Form)

Loads the Noise Gain trace data from the specified file.

Conditions Measurement view: Gain

Group Mass memory commands

Syntax `MMEMory:NOISe:GAIN:LOAD:TRACe <file_name>`

Related Commands MMEMory:NOISe:GAIN:STORe:TRACe

Arguments <file_name> = the file from which to load trace data.

Examples `MMEMORY:NOISE:GAIN:LOAD:TRACE trace1` will load the trace1 data file.

MMEMory:NOISe:GAIN:SHOW:TRACe

Show or hide the Noise Gain trace from the specified file.

Conditions Measurement view: Gain display

Group Mass memory commands

Syntax
```
MMEMory:NOISe:GAIN:SHOW:TRACe { OFF | ON | 0 | 1 }
MMEMory:NOISe:GAIN:SHOW:TRACe
```

Arguments OFF or 0 sets the trace to hide from the specified file.

ON or 1 sets the trace to show from the specified file.

Returns 0 means the trace is hidden.

1 means the trace is showing.

Examples MMEMORY:NOISE:GAIN:SHOW:TRACE? might return 1, indicating that the trace is set to show from the specified file.

MMEMory:NOISe:GAIN:STORe:TRACe (No Query Form)

Stores the Noise Gain trace data in the specified file.

Conditions Measurement view: Gain display

Group Mass memory commands

Syntax
```
MMEMory:NOISe:GAIN:STORe:TRACe <file_name>
```

Related Commands MMEMory:NOISe:GAIN:LOAD:TRACe

Arguments <file_name> = the file in which to store trace data.

Examples
```
MMEMORY:NOISE:GAIN:STORE:TRACE
```

MMEMory:Noise:LOAD:ENRCALibration (No Query Form)

Loads the specified Noise Figure ENR table.

Conditions Measurement view: Noise display

Group Mass memory commands

Syntax `MMEMory:Noise:LOAD:ENRCALibration <filename>`

Arguments <file_name> = the file from which to load ENR calibration data.

Examples `MMEMORY:NOISE:LOAD:ENRCALIBRATION`

MMEMory:Noise:LOAD:FREQTABle (No Query Form)

Loads the specified Noise Figure Frequency table.

Conditions Measurement view: Noise display

Group Mass memory commands

Syntax `MMEMory:Noise:LOAD:FREQTABle <filename>`

Arguments `<filename>` = the file from which to load the frequency table data.

Examples `MMEMORY:NOISE:LOAD:FREQTABLE`

MMEMory:Noise:STORe:ENRCALibration (No Query Form)

Saves the Noise Figure ENR table to the specified location.

Conditions Measurement view: Noise display

Group Mass memory commands

Syntax `MMEMory:Noise:STORe:ENRCALibration`

Arguments `<filename>` = the file in which to store ENR calibration data.

Examples `MMEMORY:NOISE:STORE:ENRCALIBRATION`

MMEMory:Noise:STORe:FREQTABle (No Query Form)

Saves the Noise Figure Frequency table to the specified location.

Conditions Measurement view: Noise display

Group Mass memory commands

Syntax `MMEMory:Noise:STORe:FREQTABle <filename>`

Arguments `<filename>` = the file in which to store the frequency table data.

Examples `MMEMORY:NOISE:STORE:FREQTABLE`

MMEMory:OFDM:CONSte:LOAD:TRACe (No Query Form)

Loads the OFDM constellation trace data from the specified file.

Conditions Measurement views: OFDM

Group Mass memory commands

Syntax `MMEMory:OFDM:CONSte:LOAD:TRACe <file_name>`

Arguments `<file_name> :: <string>` specifies the file from which to load the trace data.

Examples `MMEMORY:OFDM:CONSTE:LOAD:TRACE "Sample1"` loads the trace data from the file with the OFDM Constellation measurement.

MMEMory:OFDM:CONSTe:SHOW:TRACe

Enables the display of a recalled trace.

Conditions Measurement views: OFDM

Group Mass memory commands

Syntax
```
MMEMory:OFDM:CONSTe:SHOW:TRACe{ OFF | ON | 0 | 1 }
MMEMory:OFDM:CONSTe:SHOW:TRACe?
```

Arguments OFF or 0 disables display of the recalled trace.

ON or 1 enables display of the recalled trace.

Examples `MMEMORY:OFDM:CONSTE:SHOW:TRACE ON` enables the display of the recalled data loaded in the trace.

MMEMory:OFDM:CONSte:STORe:TRACe (No Query Form)

Stores the OFDM constellation measurement trace data in the specified file.

Conditions Measurement views: OFDM

Group Mass memory commands

Syntax `MMEMory:OFDM:CONSte:STORe:TRACe <file_name>`

Arguments `<file_name> :: <string>` specifies the file in which to store the trace data.

Examples `MMEMORY:OFDM:CONSTE:STORE:TRACE "Sample1"` stores the trace data in the Sample1 file in the OFDM Constellation measurement.

MMEMory:PHVTime:LOAD:TRACe (No Query Form)

Loads the Phase versus Time trace data from the specified file.

Conditions Measurement views: Phase versus Time

Group Mass memory commands

Syntax `MMEMory:PHVTime:LOAD:TRACe <file_name>`

Arguments `<file_name> :: <string>` specifies the file from which to load trace data. The file extension is .PhaseVsTime. You can omit the extension.

Examples `MMEMORY:PHVTIME:LOAD:TRACE"Sample1"` loads the Phase versus Time trace data from the *Sample1.PhaseVsTime* file.

MMEMory:PHVTime:SHOW:TRACe

Enables display of a recalled trace file in Trace<x>. The result is the same as selecting "Show recalled trace" in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions Measurement views: Phase versus Time

Group Mass memory commands

Syntax
```
MMEMory:PHVTime:SHOW:TRACe { OFF | ON | 0 | 1 }
MMEMory:PHVTime:SHOW:TRACe?
```

Related Commands MMEMory:PHVTime:LOAD:TRACe

Arguments `OFF` or `0` disables display of the recalled trace.

`ON` or `1` enables display of the recalled trace.

Examples `MMEMORY:PHVTIME:SHOW:TRACE ON` enables display of the recalled data loaded in theTrace.

MMEMory:PHVTime:STORe:TRACe (No Query Form)

Stores the Phase versus Time trace data in the specified file.

Conditions Measurement views: Phase versus Time

Group Mass memory commands

Syntax `MMEMory:PHVTime:STORe:TRACe <file_name>`

Arguments `<file_name> :: <string>` specifies the file in which to store trace data. The file extension is .PhaseVsTime. You can omit the extension.

Examples `MMEMORY:PHVTIME:STORE:TRACE"Sample1"` stores the Phase versus Time trace data in the *Sample1.PhaseVsTime* file.

MMEMory:PNOise:LOAD:TRACe<x> (No Query Form)

Loads the Phase Noise trace data from the specified file. The parameter <x> = 1 or 2; only Trace1 and Trace2 are used for Phase Noise measurement.

Conditions Measurement views: Phase noise

Group Mass memory commands

Syntax `MMEMory:PNOise:LOAD:TRACe<x> <file_name>`

Arguments `<file_name> :: <string>` specifies the file from which to load trace data.

Examples `MMEMORY:PNOISE:LOAD:TRACE1 "Wfm1"` loads Trace1 data from the Wfm1 file.

MMEMory:PNOise:SHOW:TRACe<x>

Enables display of a recalled Phase Noise trace file in Trace<x>. The result is the same as selecting “Show recalled trace” in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions Measurement views: Phase noise

Group Mass memory commands

Syntax
```
MMEMory:PNOise:SHOW:TRACe<x> { OFF | ON | 0 | 1 }
MMEMory:PNOise:SHOW:TRACe<x>?
```

Related Commands MMEMory:PNOise:LOAD:TRACe<x>

Arguments `OFF` or `0` disables display of the recalled trace.

`ON` or `1` enables display of the recalled trace.

Examples `MMEMORY:PNOISE:SHOW:TRACE1 ON` enables display of the recalled data loaded in Trace1.

MMEMory:PNOise:STORe:TRACe<x> (No Query Form)

Stores the Phase Noise trace data in the specified file. The parameter <x> = 1 or 2; only Trace1 and Trace2 are used for Phase Noise measurement.

Conditions Measurement views: Phase noise

Group Mass memory commands

Syntax `MMEMory:PNOise:STORe:TRACe<x> <file_name>`

Arguments `<file_name> :: <string>` specifies the file in which to store trace data.

Examples `MMEMORY:PNOISE:STORE:TRACE1 "WFM1"` stores Trace 1 data in the WFM1 file.

MMEMory:SEM:LOAD:TABLe (No Query Form)

Loads the Spectral Emissions Mask table containing the limits for enabled ranges from the specified file.

Conditions Measurement views: Spectral Emissions Mask

Group Mass memory commands

Syntax `MMEMory:SEM:LOAD:TABLe <file_name>`

Arguments `<file_name> :: <string>` specifies the file to load the SEM table from. The file extension is .csv. You can omit the extension.

Examples `MMEMORY:SEM:LOAD:TABLE "Table1"` loads the SEM table from the *Table1.csv* file.

MMEMory:SEM:STORe:TABLe (No Query Form)

Stores the Spectral Emissions Mask table containing the limits for enabled ranges in a specified file in the CSV format, allowing you to export the file into Microsoft Excel or other database systems.

Conditions Measurement views: Spectral Emissions Mask

Group Mass memory commands

Syntax `MMEMory:SEM:STORe:TABLe <file_name>`

Arguments `<file_name> :: <string>` specifies the file to store the SEM table in. The file extension is .csv. You can omit the extension.

Examples `MMEMORY:SEM:STORE:TABLE "Table1"` stores the SEM table in the *Table1.csv* file.

MMEMory:SGRam:LOAD:TRACe (No Query Form)

Loads the Spectrogram trace data from the specified file. No trace number is needed or allowed.

Conditions Measurement views: Spectrogram

Group Mass memory commands

Syntax `MMEMory:SGRam:LOAD:TRACe <file_name>`

Arguments `<file_name> :: <string>` specifies the file from which to load trace data. The file extension is .sogram. You can omit the extension.

Examples `MMEMORY:SGRAM:LOAD:TRACE5 "Spec2"` loads Trace5 data from the file *Spec2.sogram*.

MMEMory:SGRam:SHOW:TRACe

Enables display of a recalled trace file in Trace. The result is the same as selecting "Show recalled trace" in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions Measurement views: Spectrogram

Group Mass memory commands

Syntax

```
MMEMory:SGRam:SHOW:TRACe { OFF | ON | 0 | 1 }
MMEMory:SGRam:SHOW:TRACe?
```

Related Commands MMEMory:SGRam:LOAD:TRACe

Arguments `OFF` or `0` disables display of the recalled trace.

`ON` or `1` enables display of the recalled trace.

Examples `MMEMORY:SGRAM:SHOW:TRACE ON` enables display of the recalled data loaded in Trace.

MMEMory:SGRam:STORe:TRACe (No Query Form)

Stores the Spectrogram trace data in the specified file. No trace number is needed or allowed.

NOTE. *Only valid when the spectrum or spectrogram measurements are running.*

Conditions Measurement views: Spectrogram

Group Mass memory commands

Syntax

```
MMEMory:SGRam:STORe:TRACe <file_name>
```

Arguments `<file_name> :: <string>` specifies the file in which to store trace data. The file extension is .sogram. You can omit the extension.

Examples `MMEMORY:SGRAM:STORE:TRACE "Spec2"` stores Trace data in the *Spec2.sogram* file.

MMEMory:SPECtrum:LOAD:TRACe (No Query Form)

Loads the spectrum trace data from the specified file.

The parameter <x> = 1 to 4; Trace 5 (spectrogram) is invalid.

Conditions Measurement views: Spectrum

Group Mass memory commands

Syntax `MMEMory:SPECtrum:LOAD:TRACe <file_name>`

Arguments `<file_name> :: <string>` specifies the file from which to load trace data. The file extension is .Specan. You can omit the extension.

Examples `MMEMORY:SPECTRUM:LOAD:TRACE1 "Sample1"` loads Trace 1 data from the *Sample1*. *Specan* file.

MMEMory:SPECtrum:SHOW:TRACe<x>

Enables display of a recalled trace file in Trace<x>. The result is the same as selecting “Show recalled trace” in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions Measurement views: Spectrum

Group Mass memory commands

Syntax
```
MMEMory:SPECtrum:SHOW:TRACe<x> { OFF | ON | 0 | 1 }
MMEMory:SPECtrum:SHOW:TRACe<x>?
```

Related Commands MMEMory:SPECtrum:LOAD:TRACe

Arguments `OFF` or `0` disables display of the recalled trace.

`ON` or `1` enables display of the recalled trace.

Examples `MMEMORY:SPECTRUM:SHOW:TRACE2 ON` enables display of the recalled data loaded in Trace2.

MMEMory:SPECtrum:STORe:TRACe<x> (No Query Form)

Stores the spectrum trace data in the specified file.

The parameter <x> = 1 to 5; All traces are valid.

NOTE. *TRACe5 (spectrogram) is valid when the spectrum and spectrogram measurements are running.*

Conditions Measurement views: Spectrum

Group Mass memory commands

Syntax `MMEMory:SPECtrum:STORe:TRACe<x> <file_name>`

Arguments `<file_name> :: <string>` specifies the file in which to store trace data. The file extension is .Specan. You can omit the extension.

Examples `MMEMORY:SPECTRUM:STORE:TRACE1 "Sample1"` stores Trace 1 data in the *Sample1.Specan* file.

MMEMory:SPURious:LOAD:TABLe (No Query Form)

Loads the spurious table containing the limits for enabled ranges from the specified file.

Conditions Measurement views: Spurious

Group Mass memory commands

Syntax `MMEMory:SPURious:LOAD:TABLe <file_name>`

Arguments `<file_name> :: <string>` specifies the file to load the spurious table from. The file extension is .csv. You can omit the extension.

Examples `MMEMORY:SPURIOUS:LOAD:TRACE1 "Table1"` loads the spurious table from the *Table1.csv* file.

MMEMory:SPURious:STORe:TABLe (No Query Form)

Stores the spurious table containing the limits for enabled ranges in a specified file in the CSV (Comma Separated Values) format, allowing you to export the file into Microsoft Excel or other database systems.

Conditions Measurement views: Spurious

Group Mass memory commands

Syntax `MMEMory:SPURious:STORe:TABLe <file_name>`

Arguments `<file_name> :: <string>` specifies the file to store the spurious table in. The file extension is .csv. You can omit the extension.

Examples `MMEMORY:SPURIOUS:STORE:TABLE "Table1"` stores the spurious table in the *Table1.csv* file.

MMEMory:STORe:IQ (No Query Form)

Saves time-domain IQ waveform in the acquisition memory to a specified file.

Conditions Measurement views: All

Group Mass memory commands

Syntax `MMEMory:STORe:IQ <file_name>`

Arguments `<file_name> :: <string>` specifies the file to save IQ data. The file extension is .tiq. You can omit the extension.

Examples `MMEMORY:STORE:IQ "IQ1"` saves IQ data to the *IQ1.tiq* file.

MMEMory:STORe:IQ:CSV (No Query Form)

Saves time-domain IQ waveform in the acquisition memory to a specified file in the CSV (Comma Separated Values) format, allowing you to export the file into Microsoft Excel or other database systems.

Conditions Measurement views: All

Group Mass memory commands

Syntax `MMEMory:STORe:IQ:CSV <file_name>`

Arguments `<file_name> :: <string>` specifies the file to save IQ data. The file extension is .csv. You can omit the extension.

Examples `MMEMORY:STORE:IQ:CSV "IQ2"` saves IQ data to the *IQ2.cs(less t eq)* file.

MMEMory:STORe:IQ:MAT (No Query Form)

Saves time-domain IQ waveform in the acquisition memory to a specified file in the MATLAB format, allowing you to export the file into the MATLAB technical computing environment.

Conditions Measurement views: All

Group Mass memory commands

Syntax `MMEMory:STORe:IQ:MAT <file_name>`

Arguments `<file_name> :: <string>` specifies the file to save IQ data. The file extension is .mat. You can omit the extension.

Examples `MMEMORY:STORE:IQ:MAT "IQ3"` saves IQ data to the *IQ3.mat* file.

MMEMory:STORe:IQ:SELEct:DATA

Sets or queries the data type stored when saving acquisition data files.

Conditions Measurement views

Group Mass memory commands

Syntax `MMEMory:STORe:IQ:SELEct:DATA { IQ | SPECtra | ALL }`

NOTE. *Although the command includes IQ as part of the command, the command applies to all the data stored in the file, not only IQ data.*

Arguments `IQ` specifies that IQ data be saved in the data file.

`SPECtra` specifies that DPX spectra be saved in the data file.

`ALL` specifies that both IQ and DPX spectra be saved in the data file.

Examples `MMEMORY:STORE:IQ:SELECT:DATA SPECTRA` sets the data type to DPX Spectra.

MMEMory:STORe:IQ:SELEct:FRAMes

Sets or queries which frames are saved when saving TIQ acquisition data files.

Conditions Measurement views

Group Mass memory commands

Syntax `MMEMory:STORe:IQ:SELEct:FRAMes { ACQuisition | HISTory | SELected | ALL }`

Arguments `ACQuisition` saves the entire acquisition that contains the current frame.

`HISTory` saves the current frame.

`SELected` saves only selected frames.

`ALL` saves all acquired frames.

MMEMory:STORe:IQ:SELEct:LENGth

Sets or queries what portion of each record is saved when saving acquisition data.

Conditions Measurement views

Group Mass memory commands

Syntax
```
MMEMory:STORe:IQ:SELEct:LENGth { ENTire | LENGth }
MMEMory:STORe:IQ:SELEct:LENGth?
```

Arguments `ENTire` saves the entire record

`LENGth` saves the portion of the record being analyzed.

MMEMory:STORe:MSTate (No Query Form)

Stores the measurement parameters to a specified file in the ASCII text format for the currently selected view, allowing you to export the file into other applications.

Conditions Measurement views: All

Group Mass memory commands

Syntax
```
MMEMory:STORe:MSTate <file_name>
```

Arguments `<file_name> :: <string>` specifies the file to store the measurement parameters. The file extension is .txt. You can omit the extension.

Examples `MMEMORY:STORE:MSTATE "MSTATE1"` stores the measurement parameters to the *MSTATE1.txt* file.

MMEMory:STORe:RESults (No Query Form)

Stores the measurement results including measurement parameters and trace data to a specified file in the CSV (Comma Separated Values) format for the currently selected view, allowing you to export the file into Microsoft Excel or other database systems.

Conditions Measurement views: All

Group Mass memory commands

Syntax `MMEMory:STORe:RESults <file_name>`

Arguments `<file_name> :: <string>` specifies the file to store the measurement results. The file extension is .csv. You can omit the extension.

Examples `MMEMORY:STORE:RESULTS "RESULT1"` stores the measurement results to the *RESULT1.csv* file.

MMEMory:STORe:SCReen (No Query Form)

Stores the current display as a bitmap image file in one of several standard formats. When no format is specified, the system uses the default, .png.

Conditions Measurement views: All

Group Mass memory commands

Syntax `MMEMory:STORe:SCReen <file_name>{ .bmp | .jpg | .png }`

Arguments `<file_name> :: <string>` specifies the file to store the screen image.

Examples `MMEMORY:STORE:SCREEN "RESULT1.png"` stores the current screen image into the file*RESULT1.png*.

MMEMory:STORe:STATe (No Query Form)

Stores the instrument setup to a specified file for the currently selected view.

Conditions Measurement views: All

Group Mass memory commands

Syntax `MMEMory:STORe:STATe <file_name>`

Arguments `<file_name> :: <string>` specifies the file to store the instrument setup. The file extension is .setup. You can omit the extension.

Examples `MMEMORY:STORE:STATE "STATE1"` stores the instrument setup in the *STATE1.setup* file.

MMEMory:STORe:TRACe (No Query Form)

Stores the trace data in a specified file for the currently selected view.

Conditions Measurement views (See Table 2-26 on page 2-1071.)

Group Mass memory commands

Syntax `MMEMory:STORe:TRACe <file_name>`

Related Commands MMEMory:IQVTIME:SHOW:TRACe:I

Arguments `<file_name> :: <string>` specifies the file to store the trace data in. The file extension is named for the measurement view as shown in the table (See Table 2-26.) The trace file is not available in some views which are indicated by "NA" in the file extension column. For the views that allow multiple-trace display, select the trace to load or store using the command listed in the trace selection column.

Examples `MMEMORY:STORE:TRACE "TRACE1"` stores the trace data in the *TRACE1.Specan* file when the spectrum view is selected.

MMEMory:TOVerview:LOAD:TRACe1 (No Query Form)

Loads the trace data from a specified file into Trace1. Refer to (See Table 2-26.) for the correct file extensions for each trace type.

Conditions Measurement views

Group Mass memory commands

Syntax `MMEMory:TOVerview:LOAD:TRACe1 <file_name>`

Related Commands MMEMory:STORe:TRACe

Arguments `<file_name> :: <string>` specifies the file path and name to load the trace data from. The file extension is named for the measurement view.(See Table 2-26.). Most trace file types are compatible with only one Measurement view.

Examples `MMEMORY:TOVERVIEW:LOAD:TRACE1 "TrialB"` loads the trace data from the file*TRIALB.TOV* when the spectrum view is selected.

MMEMory:TOVerview:SHOW:TRACe1

Enables display of a recalled trace file in Trace<x>. The result is the same as selecting “Show recalled trace” in the Recall traces dialog box or selecting Show in the Trace Overview display.

Conditions Measurement views

Group Mass memory commands

Syntax
```
MMEMory:TOVerview:SHOW:TRACe1 { OFF | ON | 0 | 1 }
MMEMory:TOVerview:SHOW:TRACe1?
```

Related Commands MMEMory:TOVerview:LOAD:TRACe1

Arguments `OFF` or `0` disables display of the recalled trace.

`ON` or `1` enables display of the recalled trace.

Examples `MMEMORY:TOVERVIEW:SHOW:TRACE1 ON` enables display of the recalled data loaded in Trace2.

MMEMory:TOVerview:STORe:TRACe1 (No Query Form)

Stores the trace data for Trace1 into the specified file.

Conditions Measurement views

Group Mass memory commands

Syntax `MMEMory:TOVerview:STORe:TRACe1 <file_name>`

Related Commands MMEMory:STORe:TRACe

Arguments `<file_name> :: <string>` specifies the file to store the trace data in. The file extension is named for the measurement view as shown in the table (See Table 2-26.) The trace file is not available in some views which are indicated by "NA" in the file extension column. For the views that allow multiple-trace display, select the trace to load or store using the command listed in the trace selection column.

Examples `MMEMORY:TOVERVIEW:STORE:TRACE1 "TrialB"` stores the Trace1 data into the file*TRIALB.TOV* when a spectrum view is selected.

MMEMory:TXGain:LOAD:TRACe<x> (No Query Form)

Load the specified saved trace from a file.

Conditions Measurement view: Transmission Gain

Where <x> is the trace 1, 2, or 3.

Group Display group

Syntax `MMEMory:TXGain:LOAD:TRACe<x> <file_name>`

Arguments <file_name>::<string> specifies the file from which to load trace data.

Examples `MMEMory:TXGain:LOAD:TRACe1 "sample1"` loads the trace data from the sample1.tgt file.

MMEMory:TXGain:SHOW:TRACe<x>

Displays a saved trace instead of a live trace.

Conditions Measurement view: Transmission Gain

Where <x> is the trace 1, 2, or 3.

Group Display group

Syntax
```
MMEMory:TXGain:SHOW:TRACe<x> { OFF | ON | 0 | 1 }
MMEMory:TXGain:SHOW:TRACe<x>?
```

Arguments OFF or 0 disables display of the recalled trace.

ON or 1 enables display of the recalled trace.

Returns OFF or 0 means that display of the recalled trace is disabled.

ON or 1 means that display of the recalled trace is enabled.

Examples `MMEMory:TXGain:SHOW:TRACe2 ON` enables display of the recalled data loaded in Trace 2.

MMEMory:TXGain:STORe:TRACe<x> (No Query Form)

Saves the selected trace to a file for later recall and analysis.

Conditions Measurement view: Transmission Gain

Where <x> is the trace 1, 2, or 3.

Group Display group

Syntax
```
MMEMory:TXGain:STORe:TRACe<x> <file_name>
```

Arguments <file_name>::<string> specifies the file in which to store trace data. The file extension is .tgt. You can omit the extension.

Examples `MMEMory:TXGain:STORE:TRACe1 "Sample1"` stores Trace 1 data in the Sample1.tgt file.

MMEMory:WLAN:CRESponse:LOAD:TRACe<x> (No Query Form)

Loads the selected WLAN Channel Response trace data from the specified file.

<x> is the selected trace, 1 or 2. 1 is the magnitude trace. 2 is the phase trace.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Mass memory commands

Syntax `MMEMory:WLAN:CRESponse:LOAD:TRACe<x> <QString>`

Arguments Quoted string that specifies the file from which to load trace data.

Examples `MMEMORY:WLAN:CRESPONSE:LOAD:TRACE2 "Sample1"` loads the WLAN Channel Response trace data from the Sample1 file into Trace 2 (phase trace).

MMEMory:WLAN:CRESponse:SHOW:TRACe<x>

Enables the display of a recalled WLAN Channel Response trace.

<x> is the selected trace, 1 or 2. 1 is the magnitude trace. 2 is the phase trace.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Mass memory commands

Syntax
```
MMEMory:WLAN:CRESponse:SHOW:TRACe<x> {0|1|OFF|ON}
MMEMory:WLAN:CRESponse:SHOW:TRACe<x>?
```

Arguments `ON or 1` turns on the display of a recalled trace.

`OFF or 0` turns off the display of a recalled trace.

Examples `MMEMORY:WLAN:CRESPONSE:SHOW:TRACE2 ON` turns on the display of recalled trace 2.

MMEMory:WLAN:CRESponse:STORe:TRACe<x> (No Query Form)

Stores the WLAN Channel Response measurement trace data in the specified file.

<x> is the selected trace, 1 or 2. 1 is the magnitude trace. 2 is the phase trace.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Mass memory commands

Syntax `MMEMory:WLAN:CRESponse:STORe:TRACe<x> <QString>`

Arguments Quoted string that specifies the file in which to store the trace data.

Examples `MMEMORY:OFDM:CRESPONSE:STORE:TRACE2 "Sample1"` stores the trace data from Trace 2 into the Sample1 file.

*OPC

Generates the operation complete message in the Standard Event Status Register (SESR) when all pending operations finish. The *OPC? query places the ASCII character "1" into the output queue when all pending operations are finished. The *OPC? response is not available to read until all pending operations finish.

The *OPC command allows you to synchronize the operation of the analyzer with your application program. Refer to *Synchronizing Execution*(See page 3-11.) for the details.

Conditions Measurement views: All

Group IEEE common commands

Syntax
```
*OPC
*OPC?
```

Arguments None

*OPT? (Query Only)

Returns a list of options installed in your analyzer.

Conditions Measurement views: All

Group IEEE common commands

Syntax `*OPT?`

Arguments None

Returns The numbers of all the options installed in the analyzer, separated by commas. If no options have been installed, 0 is returned. The following table lists the options for the RSA Series analyzers.

Table 2-27: Instrument options

Option	Description
09	Enhanced real time
10	Audio analysis
11	Phase noise and jitter measurement
12	Settling time measurement (frequency and phase)
21	General purpose modulation analysis
22	Flexible OFDM analysis
23	WLAN 802.11a/b/g/j/p analysis
24	WLAN 802.11n analysis
25	WLAN 802.11ac analysis
B25	25 MHz real-time capture bandwidth
B40	40 MHz real-time capture bandwidth
B85	85 MHz real-time capture bandwidth
B16x	165 MHz real-time capture bandwidth
50	1 MHz to 3/6.2 GHz pre-amplifier (RSA5103B/RSA5106B only)
51	1 MHz to 315/26.5 GHz pre-amplifier (RSA5115B/RSA5126B only)
53	Memory 4 GB acquisition memory total
56	Removable hard disk drive (not compatible with Option 59)
59	Internal hard disk drive, incompatible with Option 56
65	Digital I and Q outputs
66	Zero-span output
6566	Digital I and Q outputs and Zero-span output
PFR	Precision frequency reference
PFR50	Precision frequency reference and internal preamp (RSA5103B/RSA5126B only)
SSD	Includes additional solid-state drive for units equipped with Option 56. Minimum disk capacity is 480 GB. The Microsoft Windows 7 operating system and instrument software is preinstalled.

Examples `*OPT?` might return `10,21,22`, indicating that Option 10, 21, and 22 are currently installed in the analyzer.

OUTPut:IF:{BANDwidth|BWIDth}

Sets or queries the IF output filter when OUTPut:IF[:STATe] is set to ON.

Conditions Measurement views: All (Option 05 only)

Group Output commands

Syntax
```
OUTPut:IF:{BANDwidth|BWIDth} { FLATtop | GAUSsian }
OUTPut:IF:{BANDwidth|BWIDth}?
```

Arguments `FLATtop` selects the flattop filter.

`GAUSsian` selects the Gaussian filter.

Examples `OUTPUT:IF:BANDWIDTH GAUSSIAN` selects the Gaussian filter for the IF output.

OUTPut:IF[:STATe]

Determines whether to turn on or off the 500 MHz IF Out on the rear panel.

Conditions Measurement views: All (Option 05 only)

Group Output commands

Syntax
```
OUTPut:IF[:STATe] { OFF | ON | 0 | 1 }
OUTPut:IF[:STATe]?
```

Related Commands OUTPut:IF:{BANDwidth|BWIDth}

Arguments `OFF` or `0` turns off IF Out.

`ON` or `1` turns on IF Out.

Examples `OUTPUT:IF:STATE ON` turns on IF Out.

OUTPut:IQ:SOURce

Sets or queries the format of the IQ output.

Conditions Measurement views: All (Option 05 only)

Group Output commands

Syntax

```
OUTPut:IQ:SOURce { NORMal | CTP }
OUTPut:IQ:SOURce?
```

Arguments `NORMal` sets the output format to normal.

`CTP` sets the output format to the test pattern.

Examples `OUTPUT:IQ:SOURCE?` might return `CTP` indicating the IQ output is a test pattern.

OUTPut:IQ[:STATe]

Determines whether to enable or disable the digital IQ output data stream from the rear panel connectors.

Conditions Measurement views: All (Option 05 only)

Group Output commands

Syntax

```
OUTPut:IQ[:STATe] { 0 | 1 | OFF | ON }
OUTPut:IQ[:STATe]?
```

Arguments `OFF` or `0` disables the digital IQ output.

`ON` or `1` enables the digital IQ output.

At *RST, this value is set to OFF.

Examples `OUTPUT:IQ:STATE ON` enables the digital IQ output.

OUTPut:NOISe[:STATe]

Determines whether to turn on or off the +28 V DC Out on the rear panel.

Conditions Measurement views: All

Group Output commands

Syntax
```
OUTPut:NOISe[:STATe] { OFF | ON | 0 | 1 }
OUTPut:NOISe[:STATe]?
```

Arguments `OFF` or `0` turns off +28 V DC Out.

`ON` or `1` turns on +28 V DC Out.

Examples `OUTPUT:NOISE:STATE ON` turns on +28 V DC Out.

OUTPut:TGen:BRIDge:MODE

Sets or queries the tracking generator bridge mode.

Conditions Requires Option 04 in a RSA500A series or RSA600A series instrument.

Group Output group

Syntax
```
OUTPut:TGen:BRIDge:MODE { REFerence | CROSSover | REFLection
| DISabled }
OUTPut:TGen:BRIDge:MODE
```

Arguments

Returns

Examples
```
OUTPUT:TGEN:BRIDGE:MODE
```

OUTPut:TGen:ENABle

Sets or queries the power state of the tracking generator.

Conditions Requires Option 04 in a RSA500A series or RSA600A series instrument.

Group Output group

Syntax
```
OUTPut:TGen:ENABle { OFF | ON | 0 | 1 }
OUTPut:TGen:ENABle?
```

Related Commands

Arguments

Returns

Examples
```
OUTPUT:TGEN:ENABLE
```

OUTPut:TGen:FREQuency

Sets or queries the tracking generator frequency in Hz. The spectrum analyzer center frequency will also be set to match (track) the tracking generator frequency.

Conditions Requires Option 04 in a RSA500A series or RSA600A series instrument.

Group Output group

Syntax
```
OUTPut:TGen:FREQuency <value>
OUTPut:TGen:FREQuency?
```

Arguments

Returns

Examples
```
OUTPUT:TGEN:FREQUENCY
```

OUTPut:TGen:INSTalled? (Query Only)

Queries for the presence of the tracking generator hardware.

Conditions Requires Option 04 in a RSA500A series or RSA600A series instrument.

Group Output group

Syntax
```
OUTPut:TGen:INSTalled?
```

Returns

Examples

```
OUTPUT:TGEN:INSTALLED?
```

OUTPut:TGen:LEVel

Sets or queries the tracking generator output level in dBm.

Conditions Requires Option 04 in a RSA500A series or RSA600A series instrument.

Group Output group

Syntax

```
OUTPut:TGen:LEVel <value>
OUTPut:TGen:LEVel?
```

Related Commands

Arguments

Returns

Examples

```
OUTPUT:TGEN:LEVEL
```

OUTPut:ZS:{BANDwidth|BWIDth}

Sets or queries the resolution bandwidth RBW filter for the Zero Span output on the rear panel connector.

Conditions Measurement views: DPX, Spectrum

Group Output commands

Syntax

```
OUTPut:ZS:{BANDwidth|BWIDth} <value>
OUTPut:ZS:{BANDwidth|BWIDth}?
```

Arguments <value> :: <NRf> specifies the resolution bandwidth.

OUTPut:ZS:{BANDwidth|BWIDth}:AUTO

Determines whether to set the Zero Span resolution bandwidth manually or automatically.

Conditions Measurement views: DPX, Spectrum

Group Output commands

Syntax

```
OUTPut:ZS:{BANDwidth|BWIDth}:AUTO { OFF | ON | 0 | 1 }
OUTPut:ZS:{BANDwidth|BWIDth}:AUTO?
```

Arguments OFF or 0 manually sets the Zero Span output.

ON or 1 automatically sets the Zero Span output.

Examples `OUTPUT:ZS:BANDWIDTH:AUTO ON` specifies that the Zero Span output will be set automatically.

OUTPut:ZS:{BANDwidth|BWIDth}:STATe

Sets or queries the state of the resolution bandwidth filter for the output connector on the rear panel.

Conditions Measurement views: DPX, Spectrum

Group Output commands

Syntax

```
OUTPut:ZS:{BANDwidth|BWIDth}:STATe { Auto | Manual | None }
OUTPut:ZS:{BANDwidth|BWIDth}:STATe?
```

Arguments `Auto` sets the resolution bandwidth filter to automatic mode.

`Manual` sets the resolution bandwidth filter to manual mode.

`None` bypasses the resolution bandwidth filter.

OUTPut:ZS:DELay? (Query Only)

Queries the delay from the input connector to the Zero Span output. The delay includes the propagation time through the entire system including filters and digital processing.

Conditions Measurement views: DPX, Spectrum

Group Output commands

Syntax

```
OUTPut:ZS:DELay?
```

Arguments None

Returns <NRf> The actual delay of the signal through the system.

OUTPut:ZS:FILTer:STATe

Sets or queries the signal path of the filter to the Zero Span output.

Conditions Measurement views: DPX, Spectrum

Group Output commands

Syntax

```
OUTPut:ZS:FILTer:STATe { AUTO | HIGH | LOW }
OUTPut:ZS:FILTer:STATe?
```

Arguments AUTO The filter path is calculated internally.

HIGH Uses the high-band 10 MHz filter.

LOW Uses the low-band 100 kHz filter.

OUTPut:ZS[:STATe]

Determines whether to enable or disable the Zero Span output from the rear panel connector.

Conditions Measurement views: DPX, Spectrum

Group Output commands

Syntax
```
OUTPut:ZS[:STATe] { OFF | ON | 0 | 1 }
OUTPut:ZS[:STATe]?
```

Arguments OFF or 0 disables the Zero Span output

ON or 1 enables the Zero Span output

Examples `OUTPUT:ZS:STATE ON` enables on the Zero Span output.

OUTPut:ZS:VOLTage

Sets or queries the voltage level at the Zero Span output on the rear panel connector.

Conditions Measurement views: DPX, Spectrum

Group Output commands

Syntax
```
OUTPut:ZS:VOLTage <value>
OUTPut:ZS:VOLTage?
```

Arguments <value> :: <NRf> voltage level at the Zero Span output connector.

OUTPut:ZS:VOLTage:AUTO

Sets or queries or queries how the output voltage is referenced.

Conditions Measurement views: DPX, Spectrum

Group Output commands

Syntax
```
OUTPut:ZS:VOLTage:AUTO { OFF | ON | 0 | 1 }
OUTPut:ZS:VOLTage:AUTO?
```

Arguments ON or 1 sets the output voltage referenced to 1.0 V.

OFF or 0 allows users to scale down the output voltage from 0.5 V to 1.0 V.

READ:ACPower? (Query Only)

Returns the Channel power and ACPR measurement results for all available channels.

Conditions Measurement views: Channel power and ACPR

Group Read commands

Syntax

```
READ:ACPower?
```

Arguments None

Returns

```
<chan_power>,<acpr_lower(1)>,<acpr_upper(1)>,
<acpr_lower(2)>,<acpr_upper(2)>,...
<acpr_lower(n)>,<acpr_upper(n)>
```

Where
<chan_power> is the average power of the main channel as the power reference in dBm. The unit can be changed by the [SENSe]:POWer:UNITs command.
<acpr_lower(n)> is the ACPR for the lower channel #n in dB.
<acpr_upper(n)> is the ACPR for the upper channel #n in dB.

The number of n depends on the setting of the [SENSe]:ACPower:CHANnel:PAIRs command.

Examples READ:ACPOWER? might return 4.227,-28.420,-23.847,-22.316,-29.225, indicating
(average power of the main channel) = 4.227 dBm,
(ACPR for the lower channel 1) = -28.420 dB,
(ACPR for the upper channel 1) = -23.847 dB,
(ACPR for the lower channel 2) = -22.316 dB, and
(ACPR for the upper channel 2) = -29.225 dB.

READ:ACPower:CHANnel:POWer? (Query Only)

Returns the average power of the main channel (power reference) in the Channel power and ACPR measurement.

Conditions Measurement views: Channel power and ACPR

Group Read commands

Syntax `READ:ACPower:CHANnel:POWer?`

Arguments None

Returns `<chan_power> ::= <NRf>` is the average power of the main channel in dBm. The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `READ:ACPOWER:CHANNEL:POWER?` might return `4.227`, indicating that the average power of the main channel is 4.227 dBm.

READ:ACPower:SPECtrum? (Query Only)

Returns spectrum trace data of the Channel power and ACPR measurement.

Conditions Measurement views: Channel power and ACPR

Group Read commands

Syntax `READ:ACPower:SPECtrum?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the spectrum trace data in dBm for the point n,
4-byte little endian floating-point format specified in IEEE 488.2.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `READ:ACPOWER:SPECTRUM?` might return `#43204xxxx...` (3204-byte data) for the spectrum trace data of the Channel power and ACPR measurement.

READ:{AM|FM|PM}? (Query Only)

Returns the trace data in the AM/FM/PM measurement.

Conditions Measurement views: AM, FM, PM

Group Read commands

Syntax `READ:{AM|FM|PM}?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
<data(n)> is the modulation factor in percent (AM), frequency deviation in Hz (FM), or phase deviation in degrees (PM) at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:AM` might return `#3156xxxx...` (156-byte data) for the AM measurement trace.

READ:AM:AMINdex? (Query Only)

Returns the modulation index which is (positive peak modulation factor - negative peak modulation factor)/2, returned as a percentage (%).

Conditions Measurement views: AM

Group Read commands

Syntax `READ:AM:AMINdex?`

Arguments None

Returns `<value> ::  <NRf>` the modulation index.

Examples `READ:AM:AMINDEX?` might return `36.48`, indicating the modulation index is 36.48%.

READ:AM:AMNegative? (Query Only)

Returns the negative peak modulation factor (–AM) in the AM measurement.

Conditions Measurement views: AM

Group Read commands

Syntax `READ:AM:AMNegative?`

Arguments None

Returns `<-AM> :: <NRf>` is the negative peak modulation factor in percent (%).

Examples `READ:AM:AMNEGATIVE?` might return `–23.4`, indicating the negative peak modulation factor is –23.4%.

READ:AM:AMPositive? (Query Only)

Returns the positive peak modulation factor (+AM) in the AM measurement.

Conditions Measurement views: AM

Group Read commands

Syntax `READ:AM:AMPositive?`

Arguments None

Returns `<+AM> :: <NRf>` is the positive peak modulation factor in percent (%).

Examples `READ:AM:AMPOSITIVE?` might return `43.8`, indicating the positive peak modulation factor is 43.8%.

READ:AM:RESult? (Query Only)

Returns the AM measurement results.

Conditions Measurement views: AM

Group Read commands

Syntax `READ:AM:RESult?`

Arguments None

Returns `<+AM>,<–AM>,<Total AM>`

Where
`<+AM> ::` `<NRf>` is the positive peak modulation factor in percent (%).
`<–AM> ::` `<NRf>` is the negative peak modulation factor in percent (%).
`<Total AM> ::` `<NRf>` is the (peak to peak modulation factor)/2 in percent (%).

Examples `READ:AM:RESULT?` might return `62.63,–50.89,56.76`.

READ:AUDio:FERRor? (Query Only)

Returns the carrier frequency error in the audio measurement.

This command is only available for FM and PM measurements.

Conditions Measurement views: Audio Summary

Group Read commands

Syntax `READ:AUDio:FERRor?`

Arguments None

Returns <value> ::= <NRf> the carrier frequency error.

Examples `READ:AUDIO:FERROR?` might return `419.9529809622` indicating that the audio frequency error is approximately 420 Hz.

READ:AUDio:FREQuency? (Query Only)

Returns the audio frequency.

Conditions Measurement views: Audio Summary

Group Read commands

Syntax `READ:AUDio:FREQuency?`

Arguments None

Returns <value> ::= <NRf> the audio frequency.

Examples `READ:AUDIO:FREQUENCY?` might return `5.8239462705E+3` indicating that the audio frequency is 5.824 kHz.

READ:AUDio:HARMonic:COUNt? (Query Only)

Returns the number of harmonics in the audio spectrum measurement.

The number of harmonics appear in the results table in the audio spectrum view.

Conditions Measurement views: Audio Spectrum

Group Read commands

Syntax `READ:AUDio:HARMonic:COUNt?`

Related Commands READ:AUDio:NHARmonic:COUNt?

Arguments None

Returns <count> ::= <NR1> the number of harmonics in the measurement view.

Examples `READ:AUDIO:HARMONIC:COUNT?` might return `3` indicating that there are three harmonics in the measurement.

READ:AUDio:HARMonic<x>:AMPLitude? (Query Only)

Returns the amplitude of the specified harmonic in the audio spectrum measurement.

The value <x> represents the specific harmonic, from 1 to the value specified by the CALCulate:AUDio:HARMonic:HNUMber command.

Conditions Measurement views: Audio Spectrum

Group Read commands

Syntax `READ:AUDio:HARMonic<x>:AMPLitude?`

Related Commands CALCulate:AUDio:HARMonic:HNUMber

Arguments None

Returns <value> ::= <NRf> the amplitude of the specified harmonic in dBc.

Examples `READ:AUDio:HARMonic3:AMPLitude?` might return `-2.861` indicating that amplitude of the third harmonic is -2.86 dBc.

READ:AUDio:HARMonic<x>:FREQuency? (Query Only)

Returns the frequency of the specified harmonic in the audio spectrum measurement.

The value <x> represents the specific harmonic, from 1 to the value specified by the CALCulate:AUDio:HARMonic:HNUMber command.

Conditions Measurement views: Audio Spectrum

Group Read commands

Syntax `READ:AUDio:HARMonic<x>:FREQuency?`

Related Commands CALCulate:AUDio:HARMonic:HNUMber

Arguments None

Returns <value> ::= <NRF> the frequency of the specified harmonic.

Examples `READ:AUDio:HARMonic3:FREQuency?` might return `17.4718007813E+3` indicating that the frequency of the third harmonic is 17.472 kHz.

READ:AUDio:HNOise? (Query Only)

Returns the difference between the current RMS modulation value and the reference value.

This information is only available when the Hum & Noise is measurement is enabled; see the [SENSe]:AUDio:HNOise:ENABle command. The reference value is stored after capturing the reference by pressing the Capture Reference button or by enabling the [SENSe]:AUDio:HNOise:REFerence command.

Conditions Measurement views: Audio Summary

Group Read commands

Syntax `READ:AUDio:HNOise?`

Related Commands [SENSe]:AUDio:HNOise:ENABle[SENSe]:AUDio:HNOise:REFerence

Arguments None

Returns <value> ::= <NRF> specifies the difference between the Hum & Noise reference and the actual signal in dB.

Examples `READ:AUDIO:HNOISE?` might return `0.00 dB` indicating that there is no difference between the Hum & Noise reference and the actual signal.

READ:AUDio:HNREFerence? (Query Only)

Returns the Hum and Noise RMS modulation reference value.

The value is stored after capturing the reference by pressing the Capture Reference button or by enabling the [SENSe]:AUDio:HNOise:REFerence command. This information is only available when the Hum & Noise measurement is enabled; see the [SENSe]:AUDio:HNOise:ENABle command.

For AM signal types, the reference represents modulation depth. For FM signal types, the reference represents frequency deviation. For PM signal types the reference represents phase. For Direct signal types the reference represents power.

Conditions Measurement views: Audio Summary

Group Read commands

Syntax `READ:AUDio:HNREFerence?`

Related Commands [SENSe]:AUDio:HNOise:ENABle[SENSe]:AUDio:HNOise:REFerence

Arguments None

Returns <value> ::= <NRf> the Hum & Noise reference frequency.

Examples `READ:AUDIO:HNREFERENCE?` might return `20.581121E+3` indicating that the Hum & Noise reference frequency is 20.58 kHz.

READ:AUDio:HPTPeak? (Query Only)

Returns the half peak-to-peak modulation excursion (where the modulation excursion depends on the signal type).

For AM signal types, the modulation excursion is "% Modulation Depth." For FM signal types, the modulation excursion is "Frequency Deviation." For PM signal types, the modulation excursion is "Phase Deviation." For Direct signal types, there is no modulation excursion; it is actually "signal excursion."

Conditions Measurement views: Audio Summary

Group Read commands

Syntax `READ:AUDio:HPTPeak?`

Arguments None

Returns <value> ::= <NRF> specifies the half peak modulation excursion.

Examples `READ:AUDIO:HPTPEAK?` might return `125.28231E+3` indicating that the half peak modulation excursion is 125.28 kHz.

READ:AUDio:MODDist? (Query Only)

Returns the modulation distortion for the audio measurement.

Conditions Measurement views: Audio Summary

Group Read commands

Syntax `READ:AUDio:MODDist?`

Arguments None

Returns <value> ::= <NRf> specifies the modulation distortion.

Examples `READ:AUDIO:MODDIST?` might return `98.6282113` indicating that the modulation distortion is 98.628%.

READ:AUDio:NHARmonic:COUNt? (Query Only)

Returns the number of non-harmonics in the audio spectrum measurement.

The number of non-harmonics appear in the results table in the audio spectrum view.

Conditions Measurement views: Audio Spectrum

Group Read commands

Syntax `READ:AUDio:NHARmonic:COUNt?`

Related Commands READ:AUDio:HARMonic:COUNt?

Arguments None

Returns <count> ::= <NR1> the number of non-harmonics in the measurement view.

Examples `READ:AUDIO:NHARMONIC:COUNT?` might return `7` indicating that there are seven harmonics in the measurement.

READ:AUDio:NHARmonic<x>:AMPLitude? (Query Only)

Returns the amplitude of the specified non-harmonic in the audio spectrum measurement.

The value <x> represents the specific non-harmonic, from 1 to the value specified by the CALCulate:AUDio:HARMonic:NHNumber command.

Conditions Measurement views: Audio Spectrum

Group Read commands

Syntax `READ:AUDio:NHARmonic<x>:AMPLitude?`

Related Commands CALCulate:AUDio:HARMonic:NHNumber

Arguments None

Returns <value> ::= <NRf> the amplitude of the specified non-harmonic in dBc.

Examples `READ:AUDio:NHARMonic3:AMPLitude?` might return `-2.861` indicating that amplitude of the third non-harmonic is -2.86 dBc.

READ:AUDio:NHARmonic<x>:FREQuency? (Query Only)

Returns the frequency of the specified non-harmonic in the audio spectrum measurement.

The value <x> represents the specific non-harmonic, from 1 to the value specified by the CALCulate:AUDio:HARMonic:NHNumber command.

Conditions Measurement views: Audio Spectrum

Group Read commands

Syntax `READ:AUDio:NHARmonic<x>:FREQuency?`

Related Commands CALCulate:AUDio:HARMonic:NHNumber

Arguments None

Returns <value> ::= <NRF> the frequency of the specified non-harmonic.

Examples `READ:AUDio:NHARMonic3:FREQuency?` might return `17.4718007813E+3` indicating that the frequency of the third non-harmonic is 17.472 kHz.

READ:AUDio:PNEGative? (Query Only)

Returns the minus peak modulation excursion (where the modulation excursion depends on the signal type).

For AM signal types, the modulation excursion is “% Modulation Depth.” For FM signal types, the modulation excursion is “Frequency Deviation.” For PM signal types, the modulation excursion is “Phase Deviation.” For Direct signal types, there is no modulation excursion; it is actually “signal excursion.”

Conditions Measurement views: Audio Summary

Group Read commands

Syntax `READ:AUDio:PNEGative?`

Arguments None

Returns

Examples `READ:AUDIO:PNEGATIVE?` might return `-196.04321E+3` indicating that the minus peak modulation excursion is -196.04 kHz.

READ:AUDio:POWer? (Query Only)

Returns the carrier power or signal power depending the signal type.

If the signal type is Direct, the returned value is the average power of the input signal. If the signal type is AM, FM, or PM, the returned value is the carrier power; the average power of the carrier signal with the modulation removed.

Conditions Measurement views: Audio Summary

Group Read commands

Syntax `READ:AUDio:POWer?`

Arguments None

Returns <value> ::= <NRf> specifies the carrier or signal power in dBm.

Examples `READ:AUDIO:POWER?` might return `-22.231123` indicating that the carrier power is -22.23 dBm.

READ:AUDio:PPOSitive? (Query Only)

Returns the positive peak modulation excursion (where the modulation excursion depends on the signal type).

For AM signal types, the modulation excursion is “% Modulation Depth.” For FM signal types, the modulation excursion is “Frequency Deviation.” For PM signal types, the modulation excursion is “Phase Deviation.” For Direct signal types, there is no modulation excursion; it is actually “signal excursion.”

Conditions Measurement views: Audio Summary

Group Read commands

Syntax `READ:AUDio:PPOSitive?`

Arguments None

Returns <value> ::= <NRf> specifies the positive peak modulation excursion.

Examples `READ:AUDIO:PPOSITIVE?` might return `215.04321E+3` indicating that the positive peak modulation excursion is -215.04 kHz.

READ:AUDio:RMS? (Query Only)

Returns the RMS modulation excursion for the audio measurement (where the modulation excursion depends on the signal type).

For AM signal types, the modulation excursion is "% Modulation Depth." For FM signal types, the modulation excursion is "Frequency Deviation." For PM signal types, the modulation excursion is "Phase Deviation." For Direct signal types, there is no modulation excursion; it is actually "signal excursion."

Conditions Measurement views: Audio Summary

Group Read commands

Syntax `READ:AUDio:RMS?`

Arguments None

Returns <value> ::= <NRf> specifies the RMS modulation excursion.

Examples `READ:AUDIO:RMS?` might return `20.575039E+3` indicating that RMS modulation excursion is 20.575 kHz.

READ:AUDio:SINad? (Query Only)

Returns the signal-to-noise and distortion for the audio measurement.

Conditions Measurement views: Audio Summary

Group Read commands

Syntax `READ:AUDio:SINad?`

Arguments None

Returns <value> ::= <NRf> specifies the RMS modulation excursion.

Examples `READ:AUDIO:SINAD?` might return `176.229024E-3` indicating that the signal-to-noise and distortion for the audio measurement is 0.18 dB.

READ:AUDio:SNOise? (Query Only)

Returns the signal level to noise level with the harmonic distortion and non-harmonic distortion components removed.

Conditions Measurement views: Audio Summary

Group Read commands

Syntax `READ:AUDio:SNOise?`

Arguments None

Returns <value> ::= <NRf> specifies the signal level to noise level.

Examples `READ:AUDIO:SNOISE?` might return `-12.8156364` indicating the signal level to noise level is -12.82 dB.

READ:AUDio:SPECtrum:TRACe<x>? (Query Only)

Returns the audio spectrum trace data for the audio measurement.

The <x> represents trace 1 for this measurement. No other traces are supported.

Conditions Measurement views: Audio Spectrum

Group Read commands

Syntax `READ:AUDio:SPECtrum:TRACe<x>?`

Arguments None

Returns #<num_digit><num_byte><date(1)><data(2)> . . . <data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the data value at the n^{th} data point in percent (%), 4-byte little endian floating point format specifed in IEEE.488.2.

Examples `READ:AUDio:SPECtrum:TRACe1?` might return `#516004xxxx ...` (16004 byte data) for the audio measurement.

READ:AUDio:THDist? (Query Only)

Returns the percentage of the total harmonic distortion in the audio measurement.

Conditions Measurement views: Audio Summary

Group Read commands

Syntax `READ:AUDio:THDist?`

Related Commands READ:AUDio:THDist:DB?

Arguments None

Returns <value> ::= <NRf> specifies the percentage of the total harmonic distortion.

Examples `READ:AUDIO:THDIST?` might return `53.332921` indicating the percentage of total harmonic distortion is 53.333%.

READ:AUDio:THDist:DB? (Query Only)

Returns the total harmonic distortion in dB in the audio measurement.

Conditions Measurement views: Audio Summary

Group Read commands

Syntax `READ:AUDio:THDist:DB?`

Related Commands READ:AUDio:THDist?

Arguments None

Returns <value> ::= <NRf> specifies the total harmonic distortion in dB.

Examples `READ:AUDIO:THDIST:DB?` might return `-5.46009` indicating that the total harmonic distortion is -5.46 dB.

READ:AUDio:TNHDist? (Query Only)

Returns the percentage of the total non-harmonic distortion in the audio measurement.

Conditions Measurement views: Audio Summary

Group Read commands

Syntax `READ:AUDio:TNHDist?`

Related Commands READ:AUDio:TNHDist:DB?

Arguments None

Returns <value> ::= <NRf> specifies the percentage of the total non-harmonic distortion.

Examples `READ:AUDIO:TNHDIST?` might return `297.332921` indicating the percentage of total non-harmonic distortion is 297.333%.

READ:AUDio:TNHDist:DB? (Query Only)

Returns the total non-harmonic distortion in dB in the audio measurement.

Conditions Measurement views: Audio Summary

Group Read commands

Syntax `READ:AUDio:TNHDist:DB?`

Related Commands READ:AUDio:TNHDist?

Arguments None

Returns <value> ::= <NRf> specifies the total non-harmonic distortion in dB.

Examples `READ:AUDIO:TNHDIST:DB?` might return `9.46009` indicating that the total harmonic distortion is 9.46 dB.

READ:AVTime:AVERage? (Query Only)

Returns the RMS (root-mean-square) value for the selected trace in the Amplitude versus Time measurement. Select the trace using the TRACe<x>:AVTime:SELect command.

Conditions Measurement views: Amplitude versus Time

Group Read commands

Syntax `READ:AVTime:AVERage?`

Arguments None

Returns <avg> :: <NRf> is the RMS amplitude in dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples READ:AVTIME:AVERAGE? might return -2.53, indicating the RMS amplitude is -2.53 dBm.

READ:AVTime:{FIRSt|SECond|THIRd|FOURth}? (Query Only)

Returns the trace data in the Amplitude versus Time measurement.

The mnemonics FIRSt, SECond, THIRd, and FOURth represent Trace 1, Trace 2, Trace 3, and Math trace, respectively. The traces can be specified by the TRACe<x>:AVTime command subgroup.

Conditions Measurement views: Amplitude versus Time

Group Read commands

Syntax READ:AVTime:{FIRSt|SECond|THIRd|FOURth}?

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the amplitude in dBm at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples READ:AVTIME:FIRST? might return #3156xxxx... (156-byte data) for Trace 1.

READ:AVTime:MAXimum? (Query Only)

Returns the maximum value for the selected trace in the Amplitude versus Time measurement. Select the trace using the TRACe<x>:AVTime:SELect command.

Conditions Measurement views: Amplitude versus Time

Group Read commands

Syntax `READ:AVTime:MAXimum?`

Related Commands READ:AVTime:MAXLocation?

Arguments None

Returns `<max> ::` `<NRf>` is the maximum Amplitude in dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `READ:AVTIME:MAXIMUM?` might return `-2.84`, indicating the maximum amplitude is -2.84 dBm.

READ:AVTime:MAXLocation? (Query Only)

Returns the time at which the amplitude is maximum for the selected trace in the Amplitude versus Time measurement. Select the trace using the TRACe<x>:AVTime:SELect command.

Conditions Measurement views: Amplitude versus Time

Group Read commands

Syntax `READ:AVTime:MAXLocation?`

Related Commands READ:AVTime:MAXimum?

Arguments None

Returns `<max_time> ::` `<NRf>` is the time at the maximum in seconds.

Examples `READ:AVTIME:MAXLOCATION?` might return `25.03E-9`, indicating the amplitude is maximum at 25.03 ns.

READ:AVTime:MINimum? (Query Only)

Returns the minimum value for the selected trace in the Amplitude versus Time measurement. Select the trace using the TRACe<x>:AVTime:SELect command.

Conditions Measurement views: Amplitude versus Time

Group Read commands

Syntax `READ:AVTime:MINimum?`

Related Commands READ:AVTime:MINLocation?

Arguments None

Returns `<min> :: <NRf>` is the minimum amplitude in dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `READ:AVTIME:MINIMUM?` might return `-57.64`, indicating the minimum amplitude is -57.64 dBm.

READ:AVTime:MINLocation? (Query Only)

Returns the time at which the amplitude is minimum for the selected trace in the Amplitude versus Time measurement. Select the trace using the TRACe<x>:AVTime:SELect command.

Conditions Measurement views: Amplitude versus Time

Group Read commands

Syntax `READ:AVTime:MINLocation?`

Related Commands READ:AVTime:MINimum?

Arguments None

Returns <min_time> :: <NRf> is the time at the minimum in seconds.

Examples READ:AVTIME:MINLOCATION? might return 450.7E-9, indicating the amplitude is minimum at 450.7 ns.

READ:AVTime:RESult? (Query Only)

Returns the measurement results for the selected trace in the Amplitude versus Time measurement. Select the trace using the TRACe<x>:AVTime:SELect command.

Conditions Measurement views: Amplitude versus Time

Group Read commands

Syntax READ:AVTime:RESult?

Arguments None

Returns <max>,<max_time>,<min>,<min_time>,<rms>

Where
<max> :: <NRf> is the maximum amplitude in dBm.
<max_time> :: <NRf> is the time at the maximum in seconds.
<min> :: <NRf> is the minimum amplitude in dBm.
<min_time> :: <NRf> is the time at the minimum in seconds.
<rms> :: <NRf> is the RMS amplitude in dBm.
The unit of amplitude can be changed by the [SENSe]:POWer:UNITs command.

Examples READ:AVTIME:RESULT? might return -2.68,48.62E-6,-82.47,22.11E-6,-8.24, indicating that the maximum amplitude is -2.68 dBm at 48.62 µs, the minimum amplitude is -82.47 dBm at 22.11 µs, and the RMS amplitude is -8.24 dBm.

READ:BIBEmissions:FTX? (Query Only)

Returns the band number corresponding to the transmitted FTX detected in the Bluetooth InBand Emission display.

Conditions Measurement views: Bluetooth InBand Emission

Group Read commands

Syntax `READ:BIBEmissions:FTX?`

Arguments None

Returns <value> ::= <NR1> the band number.

Examples `READ:BIBEMISSIONS:FTX?` might returns 39 indicating the band number corresponding to transmitted FTX.

READ:BIBEmissions:POWer? (Query Only)

Returns all channel power results from the Bluetooth InBand Emission display.

Conditions Measurement views: Bluetooth InBand Emission

Group Read commands

Syntax `READ:BIBEmissions:POWer?`

Arguments None

Returns <Band0_power>,<Band1_power>,<Band2_power>, . . . <Band78_power>

Where:
Bandx_power is the integrated power in that band. All power values are in dBm

Examples `READ:BIBEMISSIONS:POWER?` might return 4.227,-28.420,-23.847, . . . ,-29.225, indicating:
(integrated power for Band 0) = 4.227 dBm,
(Integrated power for Band 1) = -28.420 dBm,
(Integrated power for Band 2) = -23.847 dBm, and
(Integrated power for Band 78) = -29.225 dBm.

READ:BIBEmissions:RESUlts:STATus? (Query Only)

Returns the Pass/Fail result status of the Bluetooth InBand Emission display.

Conditions Measurement views: Bluetooth InBand Emission

Group Read commands

Syntax `READ:BIBEmissions:RESUlts:STATus?`

Arguments None

Examples `READ:BIBEMISSIONS:RESULTS:STATUS?` returns the Pass/Fail status result on the display.

READ:BLUEtooth:CONSte:FERRor? (Query Only)

Queries the frequency error in Hz in the Bluetooth Constellation. The frequency error is the difference between the measured carrier frequency of the signal and the user-selected center frequency.

Conditions Measurement views: Bluetooth Constellation

Group Read commands

Syntax `READ:BLUEtooth:CONSte:FERRor?`

Arguments None

Returns <freq_error> ::= <NRf> the frequency error.

Examples `READ:BLUEtooth:CONSte:FERRor?` Might return -10.7E+3, which is a frequency error of -10.7 kHz.

READ:BLUEtooth:CONSte:TRACe? (Query Only)

Queries the Bluetooth Constellation trace data.

Conditions Measurement views: Bluetooth Constellation

Group Read commands

Syntax `READ:BLUEtooth:CONSte:TRACe?`

Arguments None

Returns #<num_digit><num_byte><I(1)><Q1><I(2)><Q2> . . . <I(n)><Qn>

Where:
<num_digit> is the number of bytes in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<I(n)> and <Qn> are the normalized I- and Q-coordinate values at the nth data point. 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:BLUEtooth:CONSte:TRACe?` might return #43848xxxx . . . (3848-byte data) for Bluetooth Constellation trace.

READ:BLUEtooth:EDIagram:FDEViation? (Query Only)

Returns the frequency deviation vs. time trace data with the X values in the Bluetooth eye diagram.

Conditions Measurement views: Bluetooth eye diagram.

Group Read commands

Syntax `READ:BLUEtooth:EDIagram:FDEViation?`

Arguments None

Returns #<num_digits><num_bytes><Y(1)><X(1)><Y(2)><X(2)> . . . <Y(n)><X(n)>

Where
<num_digits> is the number of bytes in <num_bytes>.
<num_bytes> is the number of bytes of data that follow.
<Y(1)> amd <X(1)> are the normalized I- and Q-coordinate values at the nth datapoint. The 4–byte little endian floating point format is specified in IEEE 488.2.

Examples `READ:BLUEtooth:EDIagram:FDEVIiation?` might return #3160xxxx... (160-byte data) for the frequency deviation versus time trace.

READ:BLUEtooth:EDIagram:FERRor? (Query Only)

Returns the frequency error in the Bluetooth eye diagram measurement.

Conditions Measurement views: Bluetooth eye diagram.

Group Read commands

Syntax `READ:BLUEtooth:EDIagram:FERRor?`

Arguments None

Returns <freq_error> ::= <NRf> the frequency error in Hz.

Examples `READ:BLUEtooth:EDIagram:FERRor?` might return -10.7E+3, indicating the frequency error is -10.7 kHz.

READ:BLUEtooth:EDIagram:HORiz:SCALe? (Query Only)

Returns the value of the horizontal scale in the Bluetooth eye diagram.

Conditions Measurement views: Bluetooth eye diagram.

Group Read commands

Syntax `READ:BLUEtooth:EDIagram:HORiz:SCALe?`

Arguments None

Returns <value> ::= <NR3> the horizontal scale value.

Examples `READ:BLUETOOTH:EDIAGRAM:HORIZ:SCALE?` might return 2 indicating the horizontal scale value is 2 symbols.

READ:BLUEtooth:EDIagram:I? (Query Only)

Returns the I vs. time trace in the Bluetooth eye diagram.

Conditions Measurement views: Bluetooth eye diagram.

Group Read commands

Syntax `READ:BLUEtooth:EDIagram:I?`

Arguments None

Returns #<num_digits><num_bytes><Y(1)><X(1)><Y(2)><X(2)> . . . <Y(n)><X(n)>

Where
<num_digits> is the number of bytes in <num_bytes>.
<num_bytes> is the number of bytes of data that follow.
<Y(1)> amd <X(1)> are the normalized I- and Q-coordinate values at the n^{th} datapoint. The 4–byte little endian floating point format is specified in IEEE 488.2.

Examples `READ:BLUEtooth:EDIagram:I?` might return #3160xxxx... (160-byte data) for the I vs. time trace.

READ:BLUEtooth:EDIagram:Q? (Query Only)

Returns the Q vs. time trace in the Bluetooth eye diagram.

Conditions Measurement views: Bluetooth eye diagram.

Group Read commands

Syntax `READ:BLUEtooth:EDIagram:Q?`

Arguments None

Returns #<num_digits><num_bytes><Y(1)><X(1)><Y(2)><X(2)> . . . <Y(n)><X(n)>

Where
<num_digits> is the number of bytes in <num_bytes>.

<num_bytes> is the number of bytes of data that follow.
<Y(1)> amd <X(1)> are the normalized I- and Q-coordinate values at the nth datapoint. The 4–byte little endian floating point format is specified in IEEE 488.2.

Examples `READ:BLUEtooth:EDIagram:Q?` might return #3160xxxx... (160-byte data) for the Q vs. time trace.

READ:BLUEtooth:FDVTime:TRACe? (Query Only)

Returns the Bluetooth Frequency Deviation vs. Time trace data.

Conditions Measurement views: Frequency Deviation vs. Time

Group Read commands

Syntax `READ:BLUEtooth:FDVTime:TRACe?`

Arguments None

Returns #<num_digits><num_bytes><data(1)><data(2)> . . . <data(n)>

Where
<num_digits> is the number of bytes in <num_bytes>.
<num_bytes> is the number of bytes of data that follow.
<data(n)> is the frequency deviation in Hz at the nth datapoint. The 4-byte little endian floating point format is specified in IEEE 488.2.

Examples `READ:BLUEtooth:FDVTime:TRACe?` might return #3160xxxx... (160-byte data) for the trace.

READ:BLUEtooth:FREQuency:ERROr? (Query Only)

Returns the frequency error from the Bluetooth Constellation diagram.

Conditions Measurement views: Bluetooth Constellation

Group Read commands

Syntax READ:BLUEtooth:FREQuency:ERROr?

Arguments None

Returns <NR3>, the frequency error value in Hz.

Examples READ:BLUETOOTH:FREQUENCY:ERROR? might return 0.0000 indicating the frequency error is 0.0000 Hz.

READ:BLUEtooth:FREQuency:ERROr:TYPE

Returns the frequency error type in the Bluetooth Constellation display.

Conditions Measurement views: Bluetooth Constellation

Group Read commands

Syntax READ:BLUEtooth:FREQuency:ERROr:TYPE

Arguments None

Returns 1 indicates the error is automatically detected.

0 indicates the error is not automatically detected.

Examples READ:BLUETOOTH:FREQUENCY:ERROR:TYPEmight return 1 indicating the frequency error is automatically detected.

READ:BLUEtooth:FREQuency:OFFSet:DRIFt:F1FZero? (Query Only)

Returns the drift in frequency offset (f1–f0 — the frequency offset calculated in the first interval in the payload — the frequency offset calculated in the preamble) in the Bluetooth Frequency Deviation vs. Time display.

Conditions Measurement views: Frequency Deviation vs. Time

Group Read commands

Syntax `READ:BLUEtooth:FREQuency:OFFSet:DRIFt:F1FZero?`

Arguments None

Returns <NR3> the drift in frequency offset (f1–f0 — the frequency offset calculated in the first interval in the payload — the frequency offset calculated in the preamble)

Examples `READ:BLUEtooth:FREQuency:OFFSet:DRIFt:F1FZero?` might return -207.6465301514 indicating the drift in frequency offset (f1–f0 — the frequency offset calculated in the first interval in the payload — the frequency offset calculated in the preamble)

READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5? (Query Only)

Returns the maximum drift of the frequency offset in payload intervals spaced 50 µs away in the Bluetooth Frequency Deviation vs. Time display.

Conditions Measurement views: Frequency Deviation vs. Time

Group Read commands

Syntax `READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5?`

Arguments None

Returns <NR3> the maximum drift of the frequency offset in payload intervals spaced 50 µs away.

Examples `READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNF5?` might return -207.6465301514 indicating the maximum drift of the frequency offset in payload intervals spaced 50 µs away.

READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5:INTErval? (Query Only)

Returns the interval at which the maximum drift of frequency offset in a duration of 50 µs occurred.

Conditions Measurement views: Frequency Deviation vs. Time

Group Read commands

Syntax `READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFN5:INTErval?`

Arguments None

Returns <NR1> interval at which the maximum drift of frequency offset in a duration of 50 µs occurred.

Examples `READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNF5:INTErval?` might return 16, indicating the interval at which the maximum drift of frequency offset in a duration of 50 µs occurred. Here 16 refers to the 16th interval and that the drift of frequency offset calculated in the 16th interval from that calculated in the 11th interval is the maximum among those calculated 50 us away.

READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO? (Query Only)

Returns the maximum drift of the frequency offset of the intervals in the payload from the preamble packet in the Bluetooth Frequency Deviation vs. Time display.

Conditions Measurement views: Frequency Deviation vs. Time

Group Read commands

Syntax `READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO?`

Arguments None

Returns <NR3> the maximum drift of frequency offset calculated in payload intervals from the offset calculated in the preamble.

Examples `READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZero?` might return -207.6465301514, indicating that the maximum drift of frequency offset calculated in payload intervals from the offset calculated in the preamble is -207.6465301515 Hz.

READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO:INTErval? (Query Only)

Returns the interval (n) at which the maximum drift fn-f0 occurred in the Bluetooth Frequency Deviation vs. Time display. (The fn-f0 indicates the drift of the frequency offset calculated in each payload interval from the offset calculated in the preamble.)

Conditions Measurement views: Frequency Deviation vs. Time

Group Read commands

Syntax `READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZERO:INTErval?`

Arguments None

Returns <NR1> the interval at which the maximum drift (fn-f0) occurred.

Examples `READ:BLUEtooth:FREQuency:OFFSet:DRIFt:FNFZero:INTErval?` might return 16, indicating that at the 16th payload interval, the maximum drift occurred. f(16) - f(0) is the maximum drift among all f(n) -f (0).

READ:BLUEtooth:FREQuency:OFFSet:MAX? (Query Only)

Returns the maximum frequency offset in the Frequency Deviation vs. Time and the Center Frequency Offset and Drift displays.

Conditions Measurement views: Frequency Deviation vs. Time, CF Offset

Group Read commands

Syntax `READ:BLUEtooth:FREQuency:OFFSet:MAX?`

Arguments None

Returns <NR3> the maximum frequency offset value.

Examples `READ:BLUEtooth:FREQuency:OFFSet:MAX?` might return -189.4632263184, the frequency offset value.

READ:BLUEtooth:FREQuency:OFFSet:MAX:INTErval? (Query Only)

Returns the maximum frequency offset interval in the Frequency Deviation vs. Time and the Center Frequency Offset and Drift displays.

Conditions Measurement views: Frequency Deviation vs. Time

Group Read commands

Syntax `READ:BLUEtooth:FREQuency:OFFSet:MAX:INTErval?`

Arguments None

Returns <NR1> the maximum frequency offset interval.

Examples `READ:BLUEtooth:FREQuency:OFFSet:MAX:INTErval?` might return 271 indicating that the maximum drift f(n) - f(0) occurred at the 271st payload interval.

READ:BLUEtooth:FREQuency:OFFSet:PREAmble? (Query Only)

Returns the offset calculated in the preamble region in the Frequency Deviation vs. Time and the Center Frequency Offset and Drift displays.

Conditions Measurement views: Frequency Deviation vs. Time

Group Read commands

Syntax `READ:BLUEtooth:FREQuency:OFFSet:PREAmble?`

Arguments None

Returns <NR3> the offset calculated in the preamble region.

Examples `READ:BLUEtooth:FREQuency:OFFSet:PREAmble?` might return 106.4204711914, indicating the offset calculated in the preamble region.

READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:F? (Query Only)

Returns the average frequency value when the selected test pattern is Other in the Bluetooth Frequency Deviation vs. Time display.

Conditions Measurement views: Frequency Deviation vs. Time

Group Read commands

Syntax `READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:F?`

Arguments None

Returns <NRf> the average frequency.

Examples `READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:F?` might return 140.8309531250E+3, the average frequency value.

READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:F2F1ratio? (Query Only)

Returns the average ratio value for the high and low frequency deviation in the Bluetooth Frequency Deviation vs. Time display.

Conditions Measurement views: Frequency Deviation vs. Time

Group Read commands

Syntax `READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:F2F1ratio?`

Arguments None

Returns <NR1> the average ratio value.

Examples `READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:F2F1ratio?` might return ????, the average ratio value.

READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:FONE? (Query Only)

Returns the average frequency deviation for the low deviation pattern in the Bluetooth Frequency Deviation vs. Time display.

Conditions Measurement views: Frequency Deviation vs. Time

Group Read commands

Syntax `READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:FONE?`

Arguments None

Examples `READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:FONE?` might return 139.8309531250E+3, the average frequency deviation for the low deviation pattern.

READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:FTWO? (Query Only)

Returns the average frequency deviation for the high deviation pattern in the Bluetooth Frequency Deviation vs. Time display.

Conditions Measurement views: Frequency Deviation vs. Time

Group Read commands

Syntax `READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:FTWO?`

Arguments None

Examples `READ:BLUEtooth:MODUlation:CHARacteristics:AVErage:FTWO?` might return 139.8309531250E+3, the average frequency deviation for the high deviation pattern.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:COUNt? (Query Only)

Returns the number of packets analyzed to obtain average drift results from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:COUNt?`

Arguments None

Returns <NR1> the number of packets analyzed.

Examples `READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:COUNT?` might return 8, the number of packets analyzed.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero? (Query Only)

Returns the drift f1–f0 from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero?`

Arguments None

Returns <NRf> the drift f1–f0 value.

Examples `READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:F1FZERO?` might return 2.846E+3 indicating that the drift f1–f0 value is 2.846 kHz.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero:STATus? (Query Only)

Returns the Pass/Fail drift f1–f0 result status in the packet from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:F1FZero:STATus?`

Arguments None

Returns <string> the Pass/Fail result status.

Examples `READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:F1FZERO:STATUS?` might return PASS, FAIL, or NA indicating whether the drift f1–f0 is within the specified limit.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5? (Query Only)

Returns the maximum drift of the fn–fn-5 value from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5?`

Arguments None

Returns <NRf> the fn–fn-5 value.

Examples `READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:FNFN5?` might return 2.846E+3 indicating that the maximum drift fn–fn-5 value in the preamble region is -2.846 kHz.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5:STATus? (Query Only)

Returns the Pass/Fail status of the maximum fn–fn-5 drift in the packet from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFN5:STATus?`

Arguments None

Returns <string> the Pass/Fail result status.

Examples `READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:FNFN5:STATUS?` might return PASS, FAIL, or NA indicating whether the maximum drift fn–fn-5 is within the specified limit.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero? (Query Only)

Returns the maximum drift fn–f0 from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero?`

Arguments None

Returns <NRf> the fn–fn0 value.

Examples `READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:FNFZERO?` might return 4.846E+3 indicating that the drift fn–f0 value in the preamble region is 4.846 kHz.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero:STATus? (Query Only)

Returns the Pass/Fail result status of the fn–f0 maximum drift in the packet from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:FREQuency:OFFSet:DRIFt:FNFZero:STATus?`

Arguments None

Returns <string> the Pass/Fail result status.

Examples `READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:DRIFT:FNFZERO:STATUS?` might return PASS, FAIL, or NA indicating whether the maximum drift fn–f0 is within the specified limit.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX? (Query Only)

Returns the maximum frequency offset present in the packet from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX?`

Arguments None

Returns <NRf> the maximum frequency offset value in the packet.

Examples `READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:MAX?` might return 7.446E+3 indicating that the maximum frequency offset value in the packet is 7.446 kHz.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX:STATus? (Query Only)

Returns the Pass/Fail result status of the frequency offset maximum in the packet from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:FREQuency:OFFSet:MAX:STATus?`

Arguments None

Returns <string> the Pass/Fail result status.

Examples `READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:MAX:STATUS?` might return PASS, FAIL, or NA indicating whether the maximum frequency offset in the packet is within the specified limit.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble? (Query Only)

Returns the frequency offset value in the preamble region from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble?`

Arguments None

Returns <NRf> the frequency offset value in the preamble region.

Examples READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:PREAMBLE? might return 3.546E+3 indicating that the frequency offset value in the preamble region is 3.546 kHz.

READ:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble:STATus? (Query Only)

Returns the Pass/Fail result status of the frequency offset in the preamble region.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax READ:BLUEtooth:SUMMary:FREQuency:OFFSet:PREAmble:STATus?

Arguments None

Returns <string> the Pass/Fail result status.

Examples READ:BLUETOOTH:SUMMARY:FREQUENCY:OFFSET:PREAMBLE:STATUS? might return PASS, FAIL, or NA indicating whether the frequency offset in the preamble region is within the specified limit.

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:F:MAX:PERCentage:COUNt? (Query Only)

Returns the number of packets analyzed out of 10 packets for ΔFavg from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:F:MAX:PERCentage:COUNt?`

Arguments None

Returns <NR1> the number of packets analyzed.

Examples `READ:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:DELTA:F:MAX:PERCENTAGE:COUNT?` might return 3 indicating that three packets were analyzed.

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FONE:MAX:PERCentage:COUNt? (Query Only)

Returns the number of packets analyzed out of 10 packets for the ΔF1avg from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FONE:MAX:PERCentage:COUNt?`

Arguments None

Returns <NR1> the number of packets analyzed.

Examples `READ:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:DELTA:FONE:MAX:PERCENTAGE:COUNT?` might return 6 indicating that six packets were analyzed.

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage? (Query Only)

Returns the percentage of the ΔF2max% values that are above the specified limit.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage?`

Arguments None

Returns <NRf> the number of packets analyzed.

Examples `READ:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:DELTA:FTWO:MAX:PERCENTAGE?` might return 99.2 indicating that 99.2% of the deltaF2 maximum values are above the specified limit.

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage:COUNt? (Query Only)

Returns the number of packets analyzed out of 10 packets for delta F2 average from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage:COUNt?`

Arguments None

Returns <NR1> the number of packets analyzed.

Examples `READ:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:DELTA:FTWO:MAX:PERCENTAGE:COUNT?` might return 6 indicating that six packets were analyzed.

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage:STATus? (Query Only)

Returns the Pass/Fail result status of the ΔF2Max% field from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax
```
READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:DELTa:FTWO:MAX:PERCentage:STATus?
```

Arguments None

Returns <string> the Pass/Fail result status.

Examples `READ:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:DELTA:FTWO:MAX:PERCENTAGE:STATUS?` might return PASS, FAIL, or NA indicating the Pass/Fail status of the ΔF2Max% field in the Bluetooth summary.

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F? (Query Only)

Returns the ΔFavg value from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax
```
READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F?
```

Arguments None

Returns <NRf> the ΔFavg value.

Examples `READ:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:F?` might return 2.8723E+3 indicating the ΔFavg is 2.8723 kHz.

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F2F1ratio? (Query Only)

Returns the ratio of the ΔF2avg to ΔF1avg from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F2F1ratio?`

Arguments None

Returns <NRf> the value of the ΔF1avg to ΔF2avg ratio.

Examples `READ:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:F2F1RATIO?` might return 225.12E+3 indicating the average ratio is 22.5.12 kHz.

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F2F1ratio:STATus? (Query Only)

Returns the Pass/Fail results status of the ΔF2avg/ΔF1avg field from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:F2F1ratio:STATus?`

Arguments None

Returns <string> the Pass/Fail result status.

Examples `READ:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:F2F1RATIO:STATUS?` might return PASS, FAIL, or NA indicating the Pass/Fall result status of the ΔF2avg/ΔF1avg field.

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FONE? (Query Only)

Returns the delta F1 average value from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FONE?`

Arguments None

Returns <NRf> the delta F1 value.

Examples `READ:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:FONE?` might return 155.3672E+3 indicating the delta F1 average is 155.3672 kHz.

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FONE:STATus? (Query Only)

Returns the Pass/Fail results status of the delta F1 average from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FONE:STATus?`

Arguments None

Returns <string> the Pass/Fail results status.

Examples `READ:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:FONE:STATUS?` might return PASS, FAIL, or NA indicating whether the delta F1 average is within the limit.

READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FTWO? (Query Only)

Returns the delta F2 average value from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:MODUlation:CHARacteristics:AVErage:FTWO?`

Arguments None

Returns <NRf> the delta F1 value.

Examples `READ:BLUETOOTH:SUMMARY:MODULATION:CHARACTERISTICS:AVERAGE:FTWO?` might return 225.12E+3 indicating that the delta F2 value is 225.12 kHz.

READ:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage? (Query Only)

Returns the average output power from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage?`

Arguments None

Returns <NRf> the average output power value in dBm.

Examples `READ:BLUETOOTH:SUMMARY:OUTPUT:POWER:AVERAGE?` might return -29.1 indicating that the average output power is -29.1 dBm.

READ:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage:STATus? (Query Only)

Returns the Pass/Fail results status of the average output power from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:OUTPut:POWEr:AVErage:STATus?`

Arguments None

Returns <string> the Pass/Fail results status.

Examples `READ:BLUETOOTH:SUMMARY:OUTPUT:POWER:AVERAGE:STATUS?` might return PASS, FAIL, or NA indicating whether the average output power is within the specified limit.

READ:BLUEtooth:SUMMary:OUTPut:POWEr:COUNt? (Query Only)

Returns the number of packets analyzed out of 10 packets for the calculation of the output power from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:OUTPut:POWEr:COUNt?`

Arguments None

Returns <NR1> the number of packets analyzed.

Examples `READ:BLUETOOTH:SUMMARY:OUTPUT:POWER:COUNT?` might return 6 indicating that six packets were analyzed.

READ:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:COUNt? (Query Only)

Returns the number of packets analyzed out of 10 packets for calculation of the output power for EDR from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:COUNt?`

Arguments None

Returns <NR1> the number of packets analyzed.

Examples `READ:BLUETOOTH:SUMMARY:OUTPUT:POWER:EDR:COUNT?` might return 6 indicating that six packets were analyzed.

READ:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:DPSK? (Query Only)

Returns the power in the DPSK portion of the EDR burst from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:DPSK?`

Arguments None

Returns <NRf> the power in the DPSK portion of the EDR burst.

Examples `READ:BLUETOOTH:SUMMARY:OUTPUT:POWER:EDR:DPSK?` might return -29.2 indicating that the average output power is -29.2 dBm.

READ:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:GPSK? (Query Only)

Returns the power in the GFSK portion of the EDR burst from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:OUTPut:POWEr:EDR:GPSK?`

Arguments None

Returns <NRf> the power in the GFSK portion of the EDR burst.

Examples `READ:BLUETOOTH:SUMMARY:OUTPUT:POWER:EDR:GPSK?` might return -29.1 indicating that the average output power is -29.1 dBm.

READ:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK? (Query Only)

Returns the peak output power from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK?`

Arguments None

Returns <NRf> the peak output power value in dBm.

Examples READ:BLUETOOTH:SUMMARY:OUTPUT:POWER:PEAK? might return -29.3 indicating that the peak output power is -29.3 dBm.

READ:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK:STATus? (Query Only)

Returns the Pass/Fail results status of the peak output power from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax READ:BLUEtooth:SUMMary:OUTPut:POWEr:PEAK:STATus?

Arguments None

Returns <string> the Pass/Fail result status.

Examples READ:BLUETOOTH:SUMMARY:OUTPUT:POWER:PEAK:STATUS? might return PASS, FAIL, or NA indicating whether the peak output power is within the specified limit.

READ:BLUEtooth:SUMMary:PACKet:BDR:CRC? (Query Only)

Returns the Basic Rate value of the CRC field from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax READ:BLUEtooth:SUMMary:PACKet:BDR:CRC?

Arguments None

Returns <string> the value of the CRC field.

Examples `FETCh:BLUEtooth:SUMMary:PACKet:BDR:CRC?` Might return "0x9DB0" indicating the value of the CRC field in hexadecimal.

READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:ARQN? (Query Only)

Returns the value of ARQN field from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:ARQN?`

Arguments None

Returns <NR2> the value of the ARQN field.

Examples `FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:ARQN?` might return 0.000000 indicating the value of the flow field.

READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:FLOW? (Query Only)

Returns the value of Flow field from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:FLOW?`

Arguments None

Returns <NR2> the flow value.

Examples `FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:FLOW?` might return 1.000000 indicating the value of the flow field.

READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:HEC? (Query Only)

Returns the value of HEC field from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:HEC?`

Arguments None

Returns <string> the value of the HEC field.

Examples `FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:HEC?` might return "00110010" indicating the value of the HEC field in binary.

READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:LT:ADDRess? (Query Only)

Returns the LT address from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:LT:ADDRess?`

Arguments None

Returns <string> the LT address.

Examples `FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:LT:ADDR?` might return "101" indicating the LT address in binary.

READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:SEQN? (Query Only)

Returns the value of SEQN field from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:SEQN?`

Arguments None

Returns <NR2> the value of the SEQN field.

Examples `FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:SEQN?` might return 1.000000 indicating the value of the flow field.

READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:TYPE? (Query Only)

Returns the value of Type field from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:PACKet:BDR:HEADer:TYPE?`

Arguments None

Returns <string> the value of the type field.

Examples `FETCh:BLUEtooth:SUMMary:PACKet:BDR:HEADer:TYPE?` might return “00001111” indicating the type field in binary.

READ:BLUEtooth:SUMMary:PACKet:BDR:PAYLoad:LENgth? (Query Only)

Returns the Basic Rate payload length from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:PACKet:BDR:PAYLoad:LENgth?`

Arguments None

Returns <string> the value of the payload length.

Examples `FETCh:BLUEtooth:SUMMary:PACKet:BDR:PAYLoad:LENgth?` might return "0101010011" indicating the payload length value in binary.

READ:BLUEtooth:SUMMary:PACKet:BDR:PREAmble? (Query Only)

Returns the preamble of the Basic Rate packet from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:PACKet:BDR:PREAmble?`

Arguments None

Returns <string> the preamble of the packet.

Examples `FETCh:BLUEtooth:SUMMary:PACKet:BDR:PREAmble?` might return "0101" indicating the preamble is 0101 in binary.

READ:BLUEtooth:SUMMary:PACKet:BDR:SYNC:WORD? (Query Only)

Returns the sync word from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:PACKet:BDR:SYNC:WORD?`

Arguments None

Returns <string> the sync word.

Examples `FETCh:BLUEtooth:SUMMary:PACKet:BDR:SYNC:WORD?` might return "Ox4F36F2CEE85390CB" indicating the sync word in hexadecimal format.

READ:BLUEtooth:SUMMary:PACKet:BDR:TYPE? (Query Only)

Returns the packet type of the Basic Rate signal from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:PACKet:BDR:TYPE?`

Arguments None

Returns <string> the packet type.

Examples `FETCh:BLUEtooth:SUMMary:PACKet:BDR:TYPE?` might return "DH5" indicating that the DH5 packet is being analyzed.

READ:BLUEtooth:SUMMary:PACKet:BLE:ACCEss:ADDRess? (Query Only)

Returns the 32-bit Access Address from the Bluetooth Summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:PACKet:BLE:ACCEss:ADDRess?`

Arguments None

Returns <string> the 32-bit Access Address.

Examples `READ:BLUETOOTH:SUMMARY:PACKET:BLE:ACCESS:ADDRESS?` might return "0x71764129" indicating the Access Address in hexadecimal.

READ:BLUEtooth:SUMMary:PACKet:BLE:CRC? (Query Only)

Returns the value of the CRC field from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:PACKet:BLE:CRC?`

Arguments None

Returns <string> the value of the CRC field.

Examples `READ:BLUETOOTH:SUMMARY:PACKET:BLE:CRC?` might return "0x9DB0" indicating value CRC field in hexadecimal.

READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:LENGth? (Query Only)

Returns the value of the PDU length field from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:LENGth?`

Arguments None

Returns <string> the value of the length field.

Examples `READ:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:LENGTH?` might return “100101” indicating value length field in binary.

READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:RX:ADDRess? (Query Only)

Returns the PDU Rx address from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:RX:ADDRess?`

Arguments None

Returns <string> the Rx address.

Examples `READ:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:RX:ADDRESS?` might return 0.00000000 indicating Rx Adress.

READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TX:ADDRess? (Query Only)

Returns the PDU Tx address from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TX:ADDRess?`

Arguments None

Returns <string> the Tx address.

Examples `READ:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:TX:ADDRESS?` might return 0.00000000 indicating Tx Adress.

READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TYPE? (Query Only)

Returns the PDU packet type from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:PACKet:BLE:PDU:HEADer:TYPE?`

Arguments None

Returns <string> the PDU type.

Examples `READ:BLUETOOTH:SUMMARY:PACKET:BLE:PDU:HEADER:TYPE?` might return “0010” indicating the PDU type in binary.

READ:BLUEtooth:SUMMary:PACKet:BLE:PREAmble? (Query Only)

Returns the preamble of the BLE packet from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:PACKet:BLE:PREAmble?`

Arguments None

Returns <string> the preamble of the packet.

Examples `READ:BLUETOOTH:SUMMARY:PACKET:BLE:PREAMBLE?` might return "01010101" indicating the preamble is 01010101 in binary.

READ:BLUEtooth:SUMMary:PACKet:BLE:TYPE? (Query Only)

Returns the packet type from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Read commands

Syntax `READ:BLUEtooth:SUMMary:PACKet:BLE:TYPE?`

Arguments None

Returns <string> the BLE packet type.

Examples `READ:BLUETOOTH:SUMMARY:PACKET:BLE:TYPE?` might return "BLE_TEST" indicating the BLE test signal is being analyzed.

READ:CCDF? (Query Only)

Returns the CCDF measurement results.

Conditions Measurement views: CCDF

Group Read commands

Syntax `READ:CCDF?`

Arguments None

Returns `<avg_ampl>,<avg_ccdf>,<crest_factor>,<ampl_10>,<ampl_1>,<ampl_p1>,<ampl_p01>,<ampl_p001>,<ampl_p0001>`

Where
`<avg_ampl>` is the average amplitude in dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.
`<avg_ccdf>` is the average CCDF in percent.
`<crest_factor>` is the crest factor in dB.
`<ampl_10>` is the amplitude at CCDF of 10% in dB.
`<ampl_1>` is the amplitude at CCDF of 1% in dB.
`<ampl_p1>` is the amplitude at CCDF of 0.1% in dB.
`<ampl_p01>` is the amplitude at CCDF of 0.01% in dB.
`<ampl_p001>` is the amplitude at CCDF of 0.001% in dB.
`<ampl_p0001>` is the amplitude at CCDF of 0.0001% in dB.

Examples `READ:CCDF?` might return `-33.35,35.8,9.75,3.88,7.07,8.50,9.25,9.72,9.74`, indicating
(average amplitude) = -33.35 dBm,
(average CCDF) = 35.8%,
(crest factor) = 9.75 dB,
(amplitude at CCDF of 10%) = 3.88 dB,
(amplitude at CCDF of 1%) = 7.07 dB,
(amplitude at CCDF of 0.1%) = 8.50 dB,
(amplitude at CCDF of 0.01%) = 9.25 dB,
(amplitude at CCDF of 0.001%) = 9.72 dB, and
(amplitude at CCDF of 0.0001%) = 9.74 dB.

READ:CCDF:{FIRSt|SECond|THIRd}:X? (Query Only)

Returns the horizontal values of the specified trace in the CCDF measurement.

The mnemonics FIRSt, SECond, and THIRd represent Trace 1, Trace 2, and Gaussian reference curve, respectively.

NOTE. *This query is invalid when [SENSe]:CCDF:TIME:TYPE is set to CONTinuous or TOTal.*

Conditions Measurement views: CCDF

Group Read commands

Syntax `READ:CCDF:{FIRSt|SECond|THIRd}:X?`

Related Commands READ:CCDF:{FIRSt|SECond|THIRd}[:Y]?

Arguments None

Returns `#<num_digit><num_byte><x(1)><x(2)>...<x(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<x(n)>` is the horizontal value (dB) of the CCDF graph at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:CCDF:FIRST:X` might return `#41024xxxx...` (1024-byte data) for the horizontal values of Trace 1.

READ:CCDF:{FIRSt|SECond|THIRd}:XY? (Query Only)

Returns the horizontal and vertical value pairs of the specified trace in the CCDF measurement.

The mnemonics FIRSt, SECond, and THIRd represent Trace 1, Trace 2, and Gaussian reference curve, respectively.

NOTE. *This query is invalid when [SENSe]:CCDF:TIME:TYPE is set to CONTinuous or TOTal.*

Conditions Measurement views: CCDF

Group Read commands

Syntax `READ:CCDF:{FIRSt|SECond|THIRd}:XY?`

Arguments None

Returns `#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<x(n)><y(n)>` is the horizontal value (dB) and vertical value (%) pair at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:CCDF:FIRST:XY?` might return `#41024xxxx...` (1024-byte data) for the horizontal and vertical value pairs of Trace 1.

READ:CCDF:{FIRSt|SECond|THIRd}[:Y]? (Query Only)

Returns the vertical values of the specified trace in the CCDF measurement.

The mnemonics FIRSt, SECond, and THIRd represent Trace 1, Trace 2, and Gaussian reference curve, respectively.

NOTE. *This query is invalid when [SENSe]:CCDF:TIME:TYPE is set to CONTinuous or TOTal.*

Conditions Measurement views: CCDF

Group Read commands

Syntax `READ:CCDF:{FIRSt|SECond|THIRd}[:Y]?`

Arguments None

Returns `#<num_digit><num_byte><y(1)><y(2)>...<y(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.

<y(n)> is the vertical value (%) of the CCDF graph at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples READ:CCDF:FIRST:Y might return #41024xxxx... (1024-byte data) for the vertical values of Trace 1.

READ:CONSte:FERRor? (Query Only)

Returns the frequency error in Hz. The frequency error is the difference between the measured carrier frequency of the signal and the user-selected center frequency of the analyzer.

Group Read commands

Syntax READ:CONSte:FERRor?

Related Commands READ:EVM:FERRor?

Arguments None.

Returns <freq_error> :: <NRf> which is the frequency error in Hz.

Examples READ:CONSte:FERRor? might return -10.7E+3, which is a frequency error of -10.7 kHz.

READ:CONSte:RESults? (Query Only)

Returns the constellation measurement results of EVM RMS, peak and location displayed on the bottom of the screen.

Conditions Measurement views: Constellation

Group Read commands

Syntax READ:CONSte:RESults?

Arguments None

Returns

For modulation types 2|4|8|16FSK or C4FM:

`FSK_deviation_Avg_Leftmost, FSK_deviation_Avg_Rightmost`

Where
`FSK_deviation_Avg_Leftmost` is the average FSK deviation of the left-most symbol in Hz.
`FSK_deviation_Avg_Rightmost` is the average FSK deviation of the right-most symbol in Hz.

For all other valid modulation types:

`<EVM_RMS>,<EVM_peak>,<location>`

Where
`<EVM_RMS> ::` `<NRf>` is the RMS EVM in percent (%).
`<EVM_peak> ::` `<NRf>` is the peak EVM in percent (%).
`<location> ::` `<NRf>` is the peak location in symbol number.
The time unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples

`READ:CONSTE:RESULTS?` might return `2.841,3.227,68.000`, indicating that the RMS EVM is 2.841% and the peak EVM is 3.227% at symbol #68.

READ:CONSte:TRACe? (Query Only)

Returns the constellation trace data.

Conditions

Measurement views: Constellation

Group

Read commands

Syntax

`READ:CONSte:TRACe?`

Arguments

None

Returns

`#<num_digit><num_byte><I(1)><Q(1)><I(2)><Q(2)>...<I(n)><Q(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<I(n)>` and `<Q(n)>` are the normalized I- and Q-coordinate values at the n^{th} data point. 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:CONSTE:TRACE?` might return `#43848xxxx...` (3848-byte data) for the constellation trace data.

READ:DDEMod:STABle? (Query Only)

Returns the symbol table data.

Conditions Measurement views: Symbol table

Group Read commands

Syntax `READ:DDEMod:STABle?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the symbol table data for the point n,

\4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:DDEMOD:STABLE?` might return `#3512xxxx...` (512-byte data) for the symbol table.

READ:DIQVtime:FERRor? (Query Only)

Returns the frequency error in the Demod I&Q versus Time measurement.

Conditions Measurement views: Demod I&Q versus Time

Group Read commands

Syntax `READ:DIQVtime:FERRor?`

Arguments None

Returns `<freq_error> ::` `<NRf>` is the frequency error in Hz.

Examples `READ:DIQVTIME:FERROR?` might return `-10.7E+3`, indicating the frequency error is -10.7 kHz.

READ:DIQVtime:I? (Query Only)

Returns the I versus Time trace data.

Conditions Measurement views: Demod I&Q versus Time

Group Read commands

Syntax `READ:DIQVtime:I?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the I level in volts at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:DIQVTIME:I?` might return `#3160xxxx...` (160-byte data) for the I versus Time trace.

READ:DIQVtime:Q? (Query Only)

Returns the Q versus Time trace data.

Conditions Measurement views: Demod I&Q versus Time

Group Read commands

Syntax `READ:DIQVtime:Q?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the Q level in volts at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:DIQVTIME:Q?` might return `#3160xxxx...` (160-byte data) for the Q versus Time trace.

READ:DPX:DDENsity? (Query Only)

Returns the calculated density for the selected region in the DPXogram measurement.

Conditions Measurement views: DPX, Spectrum

Group Read commands

Syntax `READ:DPX:DDENsity?`

Arguments None

Returns <NR1> value.

Examples `READ:DPX:DDENSITY?` might return a DPX signal density value of 24.4802413393.

READ:DPX:DGRam:LINE:COUNt? (Query Only)

Returns the number of lines in the DPXogram measurement.

Conditions Measurement views: DPX, Spectrum

Group Read commands

Syntax `READ:DPX:DGRam:LINE:COUNt?`

Arguments None

Returns <NR1> number of lines in the measurement.

Examples READ:DPX:DGRAM:LINE:COUNT? might return 40 indicating that there were 40 lines in the measurement.

READ:DPX:DGRam:TIME[:SCALe]:OFFSet? (Query Only)

Returns the time scale offset in the DPXogram measurement.

Conditions Measurement views: DPX, Spectrum

Group Read commands

Syntax READ:DPX:DGRam:TIME[:SCALe]:OFFSet?

Arguments None

Returns <string> The time offset in the DPXogram measurement in seconds.

Examples READ:DPX:DGRAM:TIME[:SCALE]:OFFSET? might return "0.000 s" indicating that the time scale offset was 0.000 s.

READ:DPX:RESults:TRACe<x>? (Query Only)

Acquires a waveform and then returns waveform data of the specified trace <x> in the DPX spectrum measurement, where x is 1 to 6. The traces 1–4 are in the standard form. Trace 5 is the bitmap trace and its data is returned in a binary block. Trace 6 is the DPXogram trace on DPX spectrum plots. Trace 7 is the Ogram line in the DPXogram display.

Conditions Measurement views: DPX, Spectrum

Group Read commands

Syntax READ:DPX:RESults:TRACe<x>?

Arguments <NR1>

Returns For traces 1 to 4 or 6:
`#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the amplitude of the trace sample at the n^{th} data point.
<data(n) is in the 4-byte little endian floating-point format specified in IEEE 488.2.

The format is a binary block of 32 bit floating point values. Each value ranges from 0 to 1.0, where 1.0 represents 100%.

Examples `READ:DPX:RESULTS:TRACE1?` might return `#42004xxxx...` (2004-byte of data) for the waveform data of trace one (1).

READ:DPX:TRACe:AVERage? (Query Only)

Acquires a waveform and then returns waveform data of the average trace in the DPX, Spectrum measurement.

Conditions Measurement views: DPX, Spectrum

Group Read commands

Syntax `READ:DPX:TRACe:AVERage?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the waveform data of the average trace for the point n in dBm, 4-byte little endian floating-point format specified in IEEE 488.2.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `READ:DPX:TRACE:AVERAGE?` might return `#42004xxxx...` (2004-byte data) for the waveform data of the average trace.

READ:DPX:TRACe:BITMap? (Query Only)

Acquires a waveform and then returns trace waveform data of the bitmap trace in the DPX, Spectrum measurement.

Conditions Measurement views: DPX, Spectrum

Group Read commands

Syntax `READ:DPX:TRACe:BITMap?`

Arguments None

Returns The format is a binary block of 32 bit floating point values. Each value ranges from 0 to 1.0, where 1.0 represents 100%.

Examples `READ:DPX:TRACE:BITMAP?` might return `#42004xxxx...` (2004-byte data) for the waveform data of the bitmap trace.

READ:DPX:TRACe:MATH? (Query Only)

Returns waveform data of the math trace in the DPX, Spectrum measurement.

Conditions Measurement views: DPX, Spectrum

Group Read commands

Syntax `READ:DPX:TRACe:MATH?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the waveform data of the math trace for the point n in dBm, 4-byte little endian floating-point format specified in IEEE 488.2.

The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples READ:DPX:TRACE:MATH? might return #42004xxxx... (2004-byte data) for the waveform data of the math trace.

READ:DPX:TRACe:MAXimum? (Query Only)

Returns waveform data of the maximum trace in the DPX, Spectrum measurement.

Conditions Measurement views: DPX, Spectrum

Group Read commands

Syntax READ:DPX:TRACe:MAXimum?

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the waveform data of the maximum trace for the point n in dBm, 4-byte little endian floating-point format specified in IEEE 488.2.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples READ:DPX:TRACE:MAXIMUM? might return #42004xxxx... (2004-byte data) for the waveform data of the maximum trace.

READ:DPX:TRACe:MINimum? (Query Only)

Returns waveform data of the minimum trace in the DPX, Spectrum measurement.

Conditions Measurement views: DPX, Spectrum

Group Read commands

Syntax READ:DPX:TRACe:MINimum?

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the waveform data of the minimum trace for the point n in dBm. 4-byte little endian floating-point format specified in IEEE 488.2.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `READ:DPX:TRACE:MINIMUM?` might return `#42004xxxx...` (2004-byte data) for the waveform data of the minimum trace.

READ:EDIagram:FDEViation? (Query Only)

Returns the frequency deviation versus Time trace data with the X values.

Group Read commands

Syntax `READ:EDIagram:FDEViation?`

Related Commands

Returns `#<num_digit><num_byte><Y(1)><X(1)><Y(2)><X(2)>...<Y(n)><X(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<Y(n)> and <X(n)>` isfrequency deviation in Hz and time (symbols) pair at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:EDIAGRAM:FDEVIATION?` might return `#3160xxxx...` (160-byte data) for the frequency deviation versus Time trace.

READ:EDIagram:FERRor? (Query Only)

Returns the frequency error in the eye diagram measurement.

Conditions Measurement views: Eye diagram

Group Read commands

Syntax `READ:EDIagram:FERRor?`

Arguments None

Returns `<freq_error> ::` `<NRf>` is the frequency error in Hz.

Examples `READ:EDIAGRAM:FERROR?` might return `-10.7E+3`, indicating the frequency error is -10.7 kHz.

READ:EDIagram:I? (Query Only)

Returns the I versus Time trace data.

Conditions Measurement views: Eye diagram

Group Read commands

Syntax `READ:EDIagram:I?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<Y(n)><X(n)>` is the I level (normalized) and time (symbols) pair at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:EDIAGRAM:I?` might return `#3160xxxx...` (160-byte data) for the I versus Time trace.

READ:EDIagram:Q? (Query Only)

Returns the Q versus Time trace data.

Conditions Measurement views: Eye diagram

Group Read commands

Syntax `READ:EDIagram:Q?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<Y(n)><X(n)>` is the Q level (normalized) and time (symbols) pair at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:EDIAGRAM:Q?` might return `#3160xxxx...` (160-byte data) for the Q versus Time trace.

READ:EVM:FERRor? (Query Only)

Returns the frequency error in the EVM versus Time measurement.

Conditions Measurement views: EVM versus Time

Group Read commands

Syntax `READ:EVM:FERRor?`

Arguments None

Returns `<freq_error> ::` `<NRf>` is the frequency error in Hz.

Examples `READ:EVM:FERROR?` might return `-10.7E+3`, indicating the frequency error is -10.7 kHz.

READ:EVM:PEAK? (Query Only)

Returns the peak value in the EVM versus Time measurement.

Conditions Measurement views: EVM versus Time

Group Read commands

Syntax `READ:EVM:PEAK?`

Related Commands READ:EVM:PINDex?

Arguments None

Returns `<peak> ::` `<NRf>` is the peak EVM value in percent (%).

Examples `READ:EVM:PEAK?` might return `1.32`, indicating the peak EVM value is 1.32%.

READ:EVM:PINDex? (Query Only)

Returns the time at the EVM peak.

Conditions Measurement views: EVM versus Time

Group Read commands

Syntax `READ:EVM:PINDex?`

Related Commands READ:EVM:PEAK?

Arguments None

Returns `<peak_time> ::` `<NRf>` is the time at the EVM peak in symbol number. The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples `READ:EVM:PINDEX?` might return `68.000`, indicating that the EVM peak is at symbol #68.

READ:EVM:RMS? (Query Only)

Returns the RMS (Root-Mean-Square) value in the EVM versus Time measurement.

Conditions Measurement views: EVM versus Time

Group Read commands

Syntax `READ:EVM:RMS?`

Arguments None

Returns `<rms> ::  <NRf>` is the RMS EVM value in percent (%).

Examples `READ:EVM:RMS?` might return `0.582`, indicating the RMS EVM value is 0.582%.

READ:EVM:TRACe? (Query Only)

Returns the EVM versus Time trace data.

Conditions Measurement views: EVM versus Time

Group Read commands

Syntax `READ:EVM:TRACe?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.

<data(n)> is the EVM versus Time trace data for the point n in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples READ:EVM:TRACE? might return #42036xxxx... (2036-byte data) for the EVM versus Time trace.

READ:FDVTime:FERRor? (Query Only)

Returns the frequency error in the Frequency deviation versus Time measurement.

Conditions Measurement views: Frequency deviation versus Time

Group Read commands

Syntax READ:FDVTime:FERRor?

Arguments None

Returns <freq_error> :: <NRf> is the frequency error in Hz.

Examples READ:FDVTIME:FERROR? might return -10.7E+3, indicating the frequency error is -10.7 kHz.

READ:FDVTime:TRACe? (Query Only)

Returns the Frequency deviation versus Time trace data.

Conditions Measurement views: Frequency deviation versus Time

Group Read commands

Syntax READ:FDVTime:TRACe?

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the frequency deviation in Hz at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples READ:FDVTIME:TRACE? might return #3160xxxx... (160-byte data) for the Frequency deviation versus Time trace.

READ:{FM|PM}:FERRor? (Query Only)

Returns the frequency error in the Frequency modulation and Phase modulation measurements.

Conditions Measurement views: Frequency and Phase modulation

Group Read commands

Syntax READ:{FM|PM}:FERRor?

Arguments None

Returns <freq_error> :: <NRf> is the frequency error in Hz.

Examples READ:FM:FERRor? might return -10.7E+3, indicating the frequency error is -10.7 kHz.

READ:FM:PHALf? (Query Only)

Returns the half peak-peak frequency deviation (Pk-Pk/2) in the FM measurement.

Conditions Measurement views: Frequency deviation versus Time

Group Read commands

Syntax READ:FM:PHALf?

Arguments None

Returns <Pk-Pk/2> :: <NRf> is the half peak-peak frequency deviation in Hz.

Examples READ:FM:PHALF? might return 628.9E+3, indicating the half peak-peak frequency deviation is 628.9 kHz.

READ:FM:PNEGative? (Query Only)

Returns the negative peak frequency deviation (–Pk) in the FM measurement.

Conditions Measurement views: FM

Group Read commands

Syntax READ:FM:PNEGative?

Arguments None

Returns <-Pk> :: <NRf> is the negative peak frequency deviation in Hz.

Examples READ:FM:PNEGATIVE? might return –495.6E+3, indicating the negative peak frequency deviation is –495.6 kHz.

READ:FM:PPOSitive? (Query Only)

Returns the positive peak frequency deviation (+Pk) in the FM measurement.

Conditions Measurement views: FM

Group Read commands

Syntax READ:FM:PPOSitive?

Arguments None

Returns <+Pk> :: <NRf> is the positive peak frequency deviation in Hz.

Examples READ:FM:PPOSITIVE? might return 763.2E+3, indicating the positive peak frequency deviation is 763.2 kHz.

READ:FM:PTPeak? (Query Only)

Returns the peak-peak frequency deviation (Pk-Pk) in the FM measurement.

Conditions Measurement views: FM

Group Read commands

Syntax READ:FM:PTPeak?

Arguments None

Returns <Pk-Pk> :: <NRf> is the peak-peak frequency deviation in Hz.

Examples READ:FM:PTPEAK? might return 1.258E+6, indicating the peak-peak frequency deviation is 1.258 MHz.

READ:FM:RESult? (Query Only)

Returns the FM measurement results.

Conditions Measurement views: FM

Group Read commands

Syntax READ:FM:RESult?

Arguments None

Returns <+Pk>,<–Pk>,<RMS>,<Pk-Pk>,<Pk-Pk/2>

Where
<+Pk> :: <NRf> is the positive peak frequency deviation in Hz.
<-Pk> :: <NRf> is the negative peak frequency deviation in Hz.
<RMS> :: <NRf> is the RMS frequency deviation in Hz.
<Pk-Pk> :: <NRf> is the peak-peak frequency deviation in Hz.
<Pk-Pk/2> :: <NRf> is the half peak-peak frequency deviation in Hz.

Examples READ:FM:RESULT? might return
763.2E+3,-494.6E+3,271.2E+3,1.258E+6,628.9E+3.

READ:FM:RMS? (Query Only)

Returns the RMS frequency deviation in the FM measurement.

Conditions Measurement views: FM

Group Read commands

Syntax READ:FM:RMS?

Arguments None

Returns <RMS> :: <NRf> is the RMS frequency deviation in Hz.

Examples READ:FM:RMS? might return 271.2E+3, indicating the RMS frequency deviation is 271.2 kHz.

READ:{FSETtling|PSETtling}:FTTime? (Query Only)

Returns the settling time from the trigger position in seconds. The settling time is measured from the trigger point (see :FETCh:FSETtling:TRIGger:TIME).

Conditions Measurement views: Frequency and Phase Settling

Group Read commands

Syntax READ:{FSETtling|PSETtling}:FTTime?

Arguments None

Returns <value> :: <NRf> is the settling time in seconds.

Examples READ:FSETTLING:FTTIME? might return 44.8300E-6, indicating the settling time is 44.83 µs.

READ:{FSETtling|PSETtling}:MASK[:PASS]? (Query Only)

Returns whether the input signal passes the mask test.

Conditions Measurement views: Frequency and Phase Settling

Group Read commands

Syntax READ:{FSETtling|PSETtling}:MASK[:PASS]?

Arguments None

Returns 0 is returned if the signal fails the mask test.

1 is returned if the signal passes the mask test.

Examples READ:FSETTLING:MASK:PASS? might return 1, indicating the signal passed (did not exceed the mask limits).

READ:{FSETtling|PSETtling}:SETTled:FREQuency? (Query Only)

Returns the frequency at which the signal is considered settled.

Conditions Measurement views: Frequency and Phase Settling

Group Read commands

Syntax READ:{FSETtling|PSETtling}:SETTled:FREQuency?

Arguments None

Returns <value> :: <NRf> is the frequency at which the signal is settled.

Examples READ:FSETTLING:SETTLED:FREQUENCY? might return 2.44838E+9, indicating the input signal frequency at the point where the signal is considered settled is 2.44838 GHz.

READ:{FSETtling|PSETtling}:SETTled[:PASS]? (Query Only)

Returns whether the input signal is settled with the tolerance range.

Conditions Measurement views: Frequency and Phase Settling

Group Read commands

Syntax READ:{FSETtling|PSETtling}:SETTled[:PASS]?

Arguments None

Returns 0 is returned if the signal is not settled within the tolerance range.

1 is returned if the signal is settled within the tolerance range.

Examples READ:FSETTLING:SETTLED:PASS? might return 1, indicating the signal is settled within the tolerance range.

READ:{FSETtling|PSETtling}:SETTled:TIME? (Query Only)

Returns the settled time in seconds. The settled time is measured from the measurement start point.

Conditions Measurement views: Frequency and Phase Settling

Group Read commands

Syntax READ:{FSETtling|PSETtling}:SETTled:TIME?

Arguments None

Returns `<value> ::` `<NRf>` is the settled time in seconds.

Examples `READ:FSETTLING:SETTLED:TIME?` might return `299.830000E-6`, indicating the settled time is 299.83 µs.

READ:{FSETtling|PSETtling}:SLMSd[:PASS]? (Query Only)

Returns whether the input signal is settled within the specified tolerance and the signal is settled longer than the Minimum Settled Duration ([SENSe]:{FSETtling|PSETtling}:SDURation:MINimum).

Conditions Measurement views: Frequency and Phase Settling

Group Read commands

Syntax `READ:{FSETtling|PSETtling}:SLMSd[:PASS]?`

Arguments None

Returns `0` is returned if the signal is not settled within the tolerance range and minimum settled duration..

`1` is returned if the signal is settled within the tolerance range and minimum settled duration.

Examples `READ:FSETTLING:SLMSD:PASS?` might return `1`, indicating the signal is settled within the tolerance range and the minimum settled duration.

READ:{FSETtling|PSETtling}:STARt:TIME? (Query Only)

Returns the start time for the measurement in seconds. The start time is measured from the start of the analysis period.

Conditions Measurement views: Frequency and Phase Settling

Group Read commands

Syntax READ:{FSETtling|PSETtling}:STARt:TIME?

Arguments None

Returns <value> :: <NRf> is the time in seconds when the measurement started.

Examples READ:FSETTLING:START:TIME? might return 251.4300E-6, indicating the measurement started 251.43 µs after the beginning of the analysis period.

READ:{FSETtling|PSETtling}:TIME? (Query Only)

Returns the settling time in seconds. The settling time is measured from the start time (see :FETCh:FSETtling:STARt:TIME).

Conditions Measurement views: Frequency and Phase Settling

Group Read commands

Syntax READ:{FSETtling|PSETtling}:TIME?

Arguments None

Returns <value> :: <NRf> is the settling time in seconds.

Examples READ:FSETTLING:TIME? might return 48.4000E-6, indicating the settling time is 48.4 µs.

READ:{FSETtling|PSETtling}:TRACe<x>:X? (Query Only)

Returns the Frequency or Phase values of the specified trace.

The parameter <x> = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions Measurement views: Frequency and Phase Settling

Group Read commands

Syntax `READ:{FSETtling|PSETtling}:TRACe<x>:X?`

Arguments None

Returns `#<num_digit><num_byte><y(1)><y(2)>...<y(n)>`

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<y(n)> is the frequency (Hz) or phase (degrees) at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:FSETTLING:TRACE1:X?` might return `#574232xxxx...` (74232-byte data) for the frequency values of Trace 1.

READ:{FSETtling|PSETtling}:TRACe<x>:XY? (Query Only)

Returns the time and frequency or phase value pairs of the specified trace.

The parameter <x> = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions Measurement views: Frequency and Phase Settling

Group Read commands

Syntax `READ:{FSETtling|PSETtling}:TRACe<x>:XY?`

Arguments None

Returns `#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>`

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<x(n)><y(n)> is the time and frequency or phase value pair at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:FSETTLING:TRACE2:XY?` might return `#574232xxxx...` (74232-byte data) for the time and frequency pairs of the Trace 2.

READ:{FSETtling|PSETtling}:TRACe<x>[:Y]? (Query Only)

Returns the frequency or phase values of the specified trace.

The parameter <x> = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions Measurement views: Frequency and Phase Settling

Group Read commands

Syntax `READ:{FSETtling|PSETtling}:TRACe<x>[:Y]?`

Arguments None

Returns `#<num_digit><num_byte><y(1)><y(2)>...<y(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<y(n)>` is the frequency or phase value at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:FSETTLING:TRACE1:Y?` might return `#575148xxxx...` (75148-byte data) for the frequency values of Trace 1.

READ:{FSETtling|PSETtling}:TRIGger:TIME? (Query Only)

Returns the time when the trigger occurred in seconds.

Conditions Measurement views: Frequency and Phase Settling

Group Read commands

Syntax `READ:{FSETtling|PSETtling}:TRIGger:TIME?`

Arguments None

Returns `<value> :: <NRf>` is the time in seconds after the measurement start point when the trigger occurred.

Examples `READ:FSETTLING:TRIGGER:TIME?` might return `2.255E-6`, indicating the time when the trigger occurred 255 µs after the start of the measurement.

READ:{FSETtling|PSETtling}:VALue? (Query Only)

Returns the settled value in Hz for Frequency Settling and in degrees for Phase Settling.

Conditions Measurement views: Frequency and Phase Settling

Group Read commands

Syntax `READ:{FSETtling|PSETtling}:VALue?`

Arguments None

Returns `<value> ::= <NRf>` is the settling value in Hz for Frequency Settling and in degrees for Phase Settling.

Examples `READ:FSETTLING:VALUE?` might return `2.44838155E+9`, indicating the settled frequency is 2.44838 GHz.

READ:FSETtling:ERRor? (Query Only)

Returns the settled error in Hz in Frequency Settling. Only for Frequency Settling Time measurement. In Frequency Settling:

- When Target Reference is set to Auto, Settled Error = 0.

Conditions Measurement views: Frequency Settling

Group Read commands

Syntax `READ:FSETtling:ERRor?`

Arguments None

Returns <value> :: <NRf> returns the settled error in Hz.

When Target Reference is set to Auto, Settled Error = 0.

When Target Reference is set to Meas Freq:

Settled Error = Settled Frequency – (Measurement Frequency + Offset)

Examples READ:FSETTLING:ERRor? might return 0, indicating the Target Reference is set to Auto.

READ:FVTime? (Query Only)

Returns the Frequency versus Time trace data.

Conditions Measurement views: Frequency versus Time

Group Read commands

Syntax READ:FVTime?

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the frequency drift data for the point n in Hz,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples READ:FVTIME? might return #3156xxxx... (156-byte data) for the Frequency versus Time trace.

READ:FVTime:MAXimum? (Query Only)

Returns the maximum value in the Frequency versus Time measurement.

Conditions Measurement views: Frequency versus Time

Group Read commands

Syntax `READ:FVTime:MAXimum? imum`

Related Commands READ:FVTime:MAXLocation?

Arguments None

Returns `<max> ::` `<NRf>` is the maximum frequency drift in Hz.

Examples `READ:FVTIME:MAXIMUM?` might return `2.625E+6`, indicating the maximum frequency drift is 2.625 MHz.

READ:FVTime:MAXLocation? (Query Only)

Returns the time at which the frequency drift is maximum.

Conditions Measurement views: Frequency versus Time

Group Read commands

Syntax `READ:FVTime:MAXLocation?`

Related Commands READ:FVTime:MAXimum?

Arguments None

Returns `<max_time> ::` `<NRf>` is the time in seconds at which the frequency drift is maximum.

Examples `READ:FVTIME:MAXLOCATION?` might return `25.03E-9`, indicating the frequency drift is maximum at 25.03 ns.

READ:FVTime:MINimum? (Query Only)

Returns the minimum value in the Frequency versus Time measurement.

Conditions Measurement views: Frequency versus Time

Group Read commands

Syntax `READ:FVTime:MINimum?`

Related Commands READ:FVTime:MINLocation?

Arguments None

Returns `<min> ::` `<NRf>` is the minimum frequency drift in Hz.

Examples `READ:FVTIME:MINIMUM?` might return `-6.618E+6`, indicating the minimum frequency drift is -6.618 MHz.

READ:FVTime:MINLocation? (Query Only)

Returns the time at which the frequency drift is minimum.

Conditions Measurement views: Frequency versus Time

Group Read commands

Syntax `READ:FVTime:MINLocation?`

Related Commands READ:FVTime:MINimum?

Arguments None

Returns `<min_time> ::` `<NRf>` is the time in seconds at which the frequency drift is minimum.

Examples `READ:FVTIME:MINLOCATION?` might return `450.7E-9`, indicating the frequency drift is minimum at 450.7 ns.

READ:FVTime:RESult? (Query Only)

Returns the Frequency versus Time measurement results.

Conditions Measurement views: Frequency versus Time

Group Read commands

Syntax `READ:FVTime:RESult?`

Arguments None

Returns `<max>,<max_time>,<min>,<min_time>`

Where
`<max> ::` `<NRf>` is the maximum frequency drift in Hz.
`<max_time> ::` `<NRf>` is the time in seconds at which the frequency drift is maximum.
`<min> ::` `<NRf>` is the minimum frequency drift in Hz.
`<min_time> ::` `<NRf>` is the time in seconds at which the frequency drift is minimum.

Examples `READ:FVTIME:RESULT?` might return `2.625E+6,25.03E-9,-6.618E+6,450.7E-9`, indicating the maximum frequency drift is 2.625 MHz at 25.03 ns and the minimum frequency drift is -6.618 MHz at 450.7 ns.

READ:IQVTime:I? (Query Only)

Returns the I versus Time trace data.

Conditions Measurement views: RF I&Q versus Time

Group Read commands

Syntax `READ:IQVTime:I?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the I level data for the point n in volts,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:IQVTIME:I?` might return `#3160xxxx...` (160-byte data) for the I versus Time trace.

READ:IQVTime:MAXimum? (Query Only)

Returns the maximum value in the RF I&Q versus Time measurement.

Conditions Measurement views: RF I&Q versus Time

Group Read commands

Syntax `READ:IQVTime:MAXimum? imum`

Related Commands READ:IQVTime:MAXLocation?

Arguments None

Returns `<max> :: <NRf>` is the maximum I or Q level in volts.

Examples `READ:IQVTIME:MAXIMUM?` might return `1.214`, indicating the maximum I or Q level is 1.214 V.

READ:IQVTime:MAXLocation? (Query Only)

Returns the time at which the I or Q level is maximum.

Conditions Measurement views: RF I&Q versus Time

Group Read commands

Syntax `READ:IQVTime:MAXLocation?`

Related Commands READ:IQVTime:MAXimum?

Arguments None

Returns `<max_time> ::` `<NRf>` is the time in seconds at which the I or Q level is maximum.

Examples `READ:IQVTIME:MAXLOCATION?` might return `175.3E-9`, indicating the I or Q level is maximum at 175.3 ns.

READ:IQVTime:MINimum? (Query Only)

Returns the minimum value in the RF I&Q versus Time measurement.

Conditions Measurement views: RF I&Q versus Time

Group Read commands

Syntax `READ:IQVTime:MINimum?`

Related Commands READ:IQVTime:MINLocation?

Arguments None

Returns `<min> ::` `<NRf>` is the minimum I or Q level in volts.

Examples `READ:IQVTIME:MINIMUM?` might return `-370.5E-3`, indicating the minimum I or Q level is -370.5 mV.

READ:IQVTime:MINLocation? (Query Only)

Returns the time at which the I or Q level is minimum.

Conditions Measurement views: RF I&Q versus Time

Group Read commands

Syntax `READ:IQVTime:MINLocation?`

Related Commands READ:IQVTime:MINimum?

Arguments None

Returns `<min_time> ::= <NRf>` is the time in seconds at which the I or Q level is minimum.

Examples `READ:IQVTIME:MINLOCATION?` might return `450.7E-9`, indicating the I or Q level is minimum at 450.7 ns.

READ:IQVTime:Q? (Query Only)

Returns the Q versus Time trace data.

Conditions Measurement views: IQ versus Time

Group Read commands

Syntax `READ:IQVTime:Q?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the Q level data for the point n in volts,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:IQVTIME:Q?` might return `#3160xxxx...` (160-byte data) for the Q versus Time trace.

READ:IQVTime:RESult? (Query Only)

Returns the RF I&Q versus Time measurement results.

Conditions Measurement views: RF I&Q versus Time

Group Read commands

Syntax `READ:IQVTime:RESult?`

Arguments None

Returns `<max>,<max_time>,<min>,<min_time>`

Where
`<max> ::` `<NRf>` is the maximum I or Q level in volts.
`<max_time> ::` `<NRf>` is the time in seconds at which the I or Q level is maximum.
`<min> ::` `<NRf>` is the minimum I or Q level in volts.
`<min_time> ::` `<NRf>` is the time in seconds at which the I or Q level is minimum.

Examples `READ:IQVTIME:RESULT?` might return `1.214,175.3E-9,-370.5E-3,450.7E-9`, indicating the maximum I or Q level is 1.214 V at 175.3 ns and the minimum I or Q level is -370.5 mV at 450.7 ns.

READ:LTE:ACLR:MHITs:COUNt? (Query Only)

Returns the number of rows in the results table of the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Read commands

Syntax `READ:LTE:ACLR:MHITs:COUNt?`

Returns <NR1>

Examples `READ:LTE:ACLR:MHITS:COUNT?` might return 7, indicating there are seven rows in the results table.

READ:LTE:ACLR:MHITs<x>:BANDwidth? (Query Only)

Returns the bandwidth value of the channel (row) of the LTE ACLR display.

Conditions Measurement view: LTE ACLR

The parameter <x> is the row number in the LTE ACLR display results table.

Group Read commands

Syntax `READ:LTE:ACLR:MHITs<x>:BANDwidth?`

Returns <NR3>

Examples `READ:LTE:ACLR:MHITS1:BANDWIDTH?` might return 4.5150000000E+6, which indicates a bandwidth for channel 1 (row 1) of 4.515 MHz.

READ:LTE:ACLR:MHITs<x>:CHANnel:NAME? (Query Only)

Returns the channel name of the specified index in the results table of the LTE ACLR display. The minimum index will be 1 and the maximum index will be the count of the channels in the results table.

Conditions Measurement view: LTE ACLR

The parameter <x> is a positive integer and represents the row number in the LTE ACLR display results table.

Group Read commands

Syntax `READ:LTE:ACLR:MHITs<x>:CHANnel:NAME?`

Returns <string>

Examples `READ:LTE:ACLR:MHITS2:CHANNEL:NAME?` might return “B”, which is the name of channel 2 (row 2).

READ:LTE:ACLR:MHITs<x>:FREQuency:OFFSet? (Query Only)

Returns the frequency offset of the channel of the LTE ACLR display.

Conditions Measurement view: LTE ACLR

The parameter <x> is a positive integer and represents the row number in the LTE ACLR display results table.

Group Read commands

Syntax `READ:LTE:ACLR:MHITs<x>:FREQuency:OFFSet?`

Returns <NR3>

Examples `READ:LTE:ACLR:MHITS1:FREQUENCY:OFFSET?` might return 5.0000000000E+6, which indicates the frequency offset of channel 1 (row 1) is 5 MHz.

READ:LTE:ACLR:MHITs<x>:INTeg:LOWEr:ABSolute? (Query Only)

Returns the lower absolute value of the channel of the LTE ACLR display.

Conditions Measurement view: LTE ACLR

The parameter <x> is the row number in the LTE ACLR display results table.

Group Read commands

Syntax `READ:LTE:ACLR:MHITs<x>:INTeg:LOWEr:ABSolute?`

Returns <NRf>

Examples `READ:LTE:ACLR:MHITs1:INTeg:LOWEr:ABSolute?` might return -53.3921980303 which indicates lower absolute value of channel 1 (row 1) is -53.3921980303 dBm.

NOTE. *The actual units depends on the choice made from the units tab of the Analysis control panel.*

READ:LTE:ACLR:MHITs<x>:INTeg:LOWEr:RELative? (Query Only)

Returns the lower relative value of the channel of the LTE ACLR display.

Conditions Measurement view: LTE ACLR

The parameter <x> is the row number in the LTE ACLR display results table.

Group Read commands

Syntax `READ:LTE:ACLR:MHITs<x>:INTeg:LOWEr:RELative?`

Returns <NRf>

Examples `READ:LTE:ACLR:MHITS1:INTEG:LOWER:RELATIVE?` might return -53.3921980303 which indicates lower relative value of channel 1 (row 1) is -53.3921980303 dB.

READ:LTE:ACLR:MHITs<x>:INTeg:UPPEr:ABSolute? (Query Only)

Returns the upper absolute value of the channel of the LTE ACLR display.

Conditions Measurement view: LTE ACLR

The parameter <x> is a positive integer and represents the row number in the LTE ACLR display results table.

Group Read commands

Syntax `READ:LTE:ACLR:MHITs<x>:INTeg:UPPEr:ABSolute?`

Returns <NRf>

Examples `READ:LTE:ACLR:MHITS1:INTEG:UPPER:ABSOLUTE?` might return -53.3921980303 which indicates upper absolute value of channel 1 (row 1) is -53.3921980303 dBm.

NOTE. *The actual units depends on the choice made from the units tab of the Analysis control panel.*

READ:LTE:ACLR:REFerence:POWer? (Query Only)

Returns the reference power level in the LTE ACLR measurement.

Conditions Measurement view: LTE ACLR

Group Read commands

Syntax `READ:LTE:ACLR:REFerence:POWer?`

Returns <NRf>

Examples `READ:LTE:ACLR:REFERENCE:POWER?` might return –16.92, indicating that the reference power level is –16.92 dBm.

READ:LTE:ACLR:RESUlts:STATus? (Query Only)

Returns the pass or fail status of the LTE ACLR display measurement.

Conditions Measurement view: LTE ACLR

Group Read commands

Syntax `READ:LTE:ACLR:RESUlts:STATus?`

Returns 0 means measurement status is FAIL.

1 means measurement status is PASS.

Examples `READ:LTE:ACLR:RESULTS:STATUS?` might return 1, indicating that the LTE ACLR measurement status is PASS.

READ:LTE:ACLR:SPECtrum:X? (Query Only)

Returns the frequency of the spectrum trace in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Read commands

Syntax `READ:LTE:ACLR:SPECtrum:X?`

Returns #<num_digit><num_byte><x(1)><x(2)>...<x(n)>

Where,

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<x(n)> is the frequency (Hz) at the n^{th} point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:LTE:ACLR:SPECTRUM:X?` might return #516020xxxx... (16020-byte data) for the frequencies of the spectrum trace.

READ:LTE:ACLR:SPECtrum:XY? (Query Only)

Returns the frequency and amplitude pairs of the spectrum trace in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Read commands

Syntax `READ:LTE:ACLR:SPECtrum:XY?`

Returns #<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>

Where,

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<x(n)><y(n)> is the frequency (Hz) and amplitude (dBm) pair at the n^{th} point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:LTE:ACLR:SPECTRUM:XY?` might return #516020xxxx... (16020-byte data) for the frequency and amplitude pairs of the spectrum trace.

READ:LTE:ACLR:SPECtrum:Y? (Query Only)

Returns the amplitude of the spectrum trace in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Read commands

Syntax `READ:LTE:ACLR:SPECtrum:Y?`

Returns #<num_digit><num_byte><y(1)><y(2)>...<y(n)>

Where,

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<y(n)> is the frequency (Hz) at the n^{th} point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:LTE:ACLR:SPECTRUM:Y?` might return #516020xxxx... (16020-byte data) for the amplitude of the spectrum trace.

READ:LTE:CHSPectrum:OBW? (Query Only)

Returns the Occupied Bandwidth (Hz) of the LTE Channel Spectrum measurement view.

Conditions Measurement view: LTE Channel Spectrum

Group Read commands

Syntax `READ:LTE:CHSPectrum:OBW?`

Returns <NR3>

Examples `READ:LTE:CHSPECTRUM:OBW?` might return 1.0851851250E+6, indicating the Occupied Bandwidth is 1.09 MHz.

READ:LTE:CHSPectrum:POWer:CHANnel? (Query Only)

Returns the Channel Power value in LTE Channel Spectrum display. The result will be in dBm (by default) or in the units chosen from the Units tab of the Analysis Control panel.

Conditions Measurement view: LTE Channel Spectrum

Group Read commands

Syntax `READ:LTE:CHSPectrum:POWer:CHANnel?`

Returns <NRf>

Examples `READ:LTE:CHSPECTRUM:POWER:CHANNEL?` might return –14.9248560147, indicating that the Channel Power is -14.92 dBm.

READ:LTE:CHSPectrum:SPECtrum? (Query Only)

Returns spectrum trace data of the LTE Channel Spectrum measurement.

Conditions Measurement view: LTE Channel Spectrum

Group Read commands

Syntax `READ:LTE:CHSPectrum:SPECtrum?`

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of the data that follow.

<data(n)> is the amplitude in dBm at the n^{th} data poin, 4-byte little endian floating point format specified in IEEE 488.2.

Examples `READ:LTE:CHSPECTRUM:SPECTRUM?` might return #43204 xxxx... (3204-byte data) for the spectrum trace data of the LTE Channel Spectrum measurement.

READ:LTE:CONSte:CELL:ID? (Query Only)

Queries the Cell ID value LTE Constellation display.

Conditions Measurement view: LTE Constellation

Group Read commands

Syntax `READ:LTE:CONSte:CELL:ID?`

Returns <NR1>

Examples `READ:LTE:CONSTE:CELL:ID? ?` might return 112, indicating CELLID measurement number is 112.

READ:LTE:CONSte:FREQuency:ERRor? (Query Only)

Queries Frequency Error value in Hz for the LTE Constellation display.

Conditions Measurement view: LTE Constellation

Group Read commands

Syntax `READ:LTE:CONSte:FREQuency:ERRor?`

Returns <NRf>

Examples `READ:LTE:CONSTE:FREQUENCY:ERROR?` might return -71.9780578613, indicating frequency error value is -71.9780578613 Hz.

READ:LTE:CONSte:GROUp:ID? (Query Only)

Queries the Group ID value for the LTE Constellation display.

Conditions Measurement view: LTE Constellation

Group Read commands

Syntax `READ:LTE:CONSte:GROUp:ID?`

Returns <NR1>

Examples `READ:LTE:CONSTE:GROUP:ID?` ? might return 160, indicating Group ID measurement number is 160.

READ:LTE:CONSte:SECTor:ID? (Query Only)

Queries the Sector ID value for the LTE Constellation display.

Conditions Measurement view: LTE Constellation

Group Read commands

Syntax `READ:LTE:CONSte:SECTor:ID?`

Returns <NRf>

Examples `READ:LTE:CONSTE:SECTOR:ID?` might return 2, indicating the Sector ID value is 2.

READ:LTE:PVTime:OFFSlot:POWer? (Query Only)

Returns the TOff power measurement value in dBm/MHz for the LTE Power vs Time display.

Conditions Measurement view: LTE Power vs Time

Group Read commands

Syntax `READ:LTE:PVTime:OFFSlot:POWer?`

Returns <NRf>

Examples `READ:LTE:PVTIME:OFFSLOT:POWER?` might return -76.11514587403, indicating the offslot power is -76.11514 dBm/MHz.

READ:LTE:PVTime:RESUlts:STATus

Returns the pass or fail status for the LTE Power vs Time measurement..

Conditions Measurement view: LTE Power vs Time

Group Read commands

Syntax `READ:LTE:PVTime:RESUlts:STATus`

Returns 0: means measurement status is FAIL.

1: means measurement status is PASS.

Examples `READ:LTE:PVTIME:RESULTS:STATUS` might return 1, indicating the measurement status is PASS.

READ:LTE:PVTime:TRACe:X? (Query Only)

Returns the horizontal values (time in seconds) for the LTE Power vs. Time trace.

Conditions Measurement view: LTE Power vs Time

Group Read commands

Syntax `READ:LTE:PVTime:TRACe:X?`

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where,

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the n^{th} point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:LTE:PVTIME:TRACE:X?` might return #43204 (3204-byte data), which represents the time in seconds (horizontal values).

READ:LTE:PVTime:TRACe:XY? (Query Only)

Returns the horizontal value (time in seconds) and vertical value (power) for the LTE Power vs. Time trace.

Conditions Measurement view: LTE Power vs Time

Group Read commands

Syntax `READ:LTE:PVTime:TRACe:XY?`

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where,

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the n^{th} point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:LTE:PVTIME:TRACE:XY?` might return #43204 (3204-byte data), which represents the horizontal value (time in seconds) and vertical value (power) pair at the n^{th} data point.

READ:LTE:PVTime:TRACe:Y? (Query Only)

Returns the vertical values (power) for the LTE Power vs. Time trace.

Conditions Measurement view: LTE Power vs Time

Group Read commands

Syntax `READ:LTE:PVTime:TRACe:Y?`

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where,

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the n^{th} point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:LTE:PVTIME:TRACE:Y?` might return #43204 (3204-byte data), which represents the vertical values (power).

READ:MCPower:ADJacent:CHANnels? (Query Only)

Returns the power of adjacent channels in order of increasing frequency.

Conditions Measurement views: MCPR

Group Read commands

Syntax `READ:MCPower:ADJacent:CHANnels?`

Arguments None

Returns

<acpr_lower(n)>,...<acpr_lower(2)>,<acpr_lower(1)>,
<acpr_upper(1)>,<acpr_upper(2)>,...<acpr_upper(n)>

Where
<acpr_lower(n)> is the ACPR for the lower channel #n in dB.
<acpr_upper(n)> is the ACPR for the upper channel #n in dB.

To add a pair of upper and lower adjacent channels, use the [SENSe]:MCPower:CHANnel:ADJacent:ADD command.

Examples

READ:MCPOWER:ADJACENT:CHANNELS? might return -4.420,-4.847,-4.316,-4.225, indicating
(ACPR for the lower channel 2) = -4.420 dB,
(ACPR for the lower channel 1) = -4.847 dB,
(ACPR for the upper channel 1) = -4.316 dB, and
(ACPR for the upper channel 2) = -4.225 dB.

READ:MCPower:CHANnel:POWer? (Query Only)

Returns the reference power in the MCPR measurement.

Conditions

Measurement views: MCPR

Group

Read commands

Syntax

READ:MCPower:CHANnel:POWer?

Arguments

None

Returns

<ref_power>:<NRf> is the reference power in dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

To select the power reference, use the [SENSe]:MCPower:RCHannels commands.

Examples

READ:MCPOWER:CHANNEL:POWER? might return 4.227, indicating that the reference power is 4.227 dBm.

READ:MCPower:MAIN:CHANnels? (Query Only)

Returns the power of main channels in order of increasing frequency.

Conditions Measurement views: MCPR

Group Read commands

Syntax `READ:MCPower:MAIN:CHANnels?`

Related Commands [:SENSe]:MCPower:CHANnel:MAIN commands

Arguments None

Returns `<power_main(1)>,<power_main(2)>,...<power_main(n)>`

Where
`<power_main(n)>` is the power of main channel #n in dBm.
The unit can be changed by the[SENSe]:POWer:UNITs command.

To specify the main channels, use the [SENSe]:MCPower:CHANnel:MAIN commands.

Examples `READ:MCPOWER:MAIN:CHANNELS?` might return `-2.420,-2.847,-2.316,-2.225`, indicating
(power of the main channel 1) = -2.420 dBm,
(power of the main channel 2) = -2.847 dBm,
(power of the main channel 3) = -2.316 dBm, and
(power of the main channel 4) = -2.225 dBm.

READ:MCPower:SPECtrum? (Query Only)

Returns spectrum trace data of the MCPR measurement.

Conditions Measurement views: MCPR

Group Read commands

Syntax `READ:MCPower:SPECtrum?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the spectrum trace data in dBm for the point n,
4-byte little endian floating-point format specified in IEEE 488.2.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples READ:MCPOWER:SPECTRUM? might return #43204xxxx... (3204-byte data) for the spectrum trace data of the MCPR measurement.

READ:MERRor:FERRor? (Query Only)

Returns the frequency error in the Magnitude error versus Time measurement.

Conditions Measurement views: Magnitude error versus Time

Group Read commands

Syntax READ:MERRor:FERRor?

Arguments None

Returns <freq_error> :: <NRf> is the frequency error in Hz.

Examples READ:MERROR:FERROR? might return -10.7E+3, indicating the frequency error is -10.7 kHz.

READ:MERRor:PEAK? (Query Only)

Returns the peak value in the Magnitude error versus Time measurement.

Conditions Measurement views: Magnitude error versus Time

Group Read commands

Syntax READ:MERRor:PEAK?

Related Commands READ:MERRor:PINDex?

Arguments None

Returns <peak> :: <NRf> is the peak magnitude error in percent (%).

Examples READ:MERROR:PEAK? might return 1.57, indicating the peak magnitude error is 1.57%.

READ:MERRor:PINDex? (Query Only)

Returns the time at the magnitude error peak.

Conditions Measurement views: Magnitude error versus Time

Group Read commands

Syntax READ:MERRor:PINDex?

Related Commands READ:MERRor:PEAK?

Arguments None

Returns <peak_time> :: <NRf> is the time at the magnitude error peak in symbol number. The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples READ:MERROR:PINDEX? might return 68.000, indicating that the magnitude error peak is at symbol #68.

READ:MERRor:RMS? (Query Only)

Returns the RMS (Root-Mean-Square) value in the Magnitude error versus Time measurement.

Conditions Measurement views: Magnitude error versus Time

Group Read commands

Syntax `READ:MERRor:RMS?`

Arguments None

Returns `<rms> ::` `<NRf>` is the RMS magnitude error in percent (%).

Examples `READ:MERROR:RMS?` might return `0.382`, indicating the magnitude error is 0.382% RMS.

READ:MERRor:TRACe? (Query Only)

Returns the Magnitude error versus Time trace data.

Conditions Measurement views: Magnitude error versus Time

Group Read commands

Syntax `READ:MERRor:TRACe?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the magnitude error data for the point n in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:MERROR:TRACE?` might return `#42036xxxx...` (2036-byte data) for the Magnitude error versus Time trace.

READ:NOISe:TABLe:VALue? (Query Only)

Returns the noise table data. A total of 7 values per frequency (28 bytes per frequency) is returned. The number of frequencies returned depends on the settings.

Conditions Measurement view: Noise display

Group Read commands

Syntax `READ:NOISe:TABLe:VALue?`

Returns Returns an array of floats (size of 4 bytes each) defined as:

- First frequency gain, noise figure, noise factor, y factor, noise temperature, power hot, power cold
- Second frequency gain, noise figure, noise factor, y factor, noise temperature, power hot, power cold

Examples `READ:NOISE:TABLE:VALUE?`

READ:OBWidth:BOBW:XDBBandwidth? (Query Only)

Returns the difference between the higher and lower frequency points corresponding to the value where it is X dB less from the value at the center frequency.

Set the measurement direction using the appropriate inward or outward command before issuing this command.

Conditions Measurement views: Bluetooth 20 dB Bandwith

Group Read commands

Syntax `READ:OBWidth:BOBW:XDBBandwidth?`

Related Commands READ:OBWidth:BOBW:XDBMbandwidth:IN:BANDwidth?, READ:OBWidth:BOBW:XDBMbandwidth:OUT:BANDwidth?

Arguments None

Returns <value> ::= <NRf> the X dBm bandwidth in Hz.

Examples `READ:OBWIDTH:BOBW:XDBBANDWIDTH?` might return 89.12000E+3 indicating the difference between the higher and lower frequency points is 89.12 kHz.

READ:OBWidth:BOBW:XDBMbandwidth:IN:BANDwidth? (Query Only)

Returns the difference between the higher and lower frequency points corresponding to the value which it is the X dBm value measured inwards (coming from the ends to the center frequency).

Conditions Measurement views: Bluetooth 20 dB Bandwith

Group Read commands

Syntax `READ:OBWidth:BOBW:XDBMbandwidth:IN:BANDwidth?`

Related Commands READ:OBWidth:BOBW:XDBBandwidth?, READ:OBWidth:BOBW:XDBMbandwidth:IN:RIGHt:FREQuency?

Arguments None

Returns <value> ::= <NRf> the input X dBm value in Hz.

Examples `READ:OBWIDTH:BOBW:XDBMBANDWIDTH:IN:BANDWIDTH?` might return 1.2518400000E+6 indicating that the bandwidth corresponding to the input XdBm vaue is 1.2518 MHz measured in the inward direction.

READ:OBWidth:BOBW:XDBMbandwidth:IN:LEFT:FREQuency? (Query Only)

Returns the frequency corresponding to the X dBm value to the left of the center frequency measured in the inward direction.

Conditions Measurement views: Bluetooth 20 dB Bandwith

Group Read commands

Syntax `READ:OBWidth:BOBW:XDBMbandwidth:IN:LEFT:FREQuency?`

Related Commands READ:OBWidth:BOBW:XDBMbandwidth:IN:RIGHt:FREQuency?

Arguments None

Returns <value> ::= <NRf> the left frequency in Hz.

Examples `READ:OBWIDTH:BOBW:XDBMBANDWIDTH:IN:LEFT:FREQUENCY?` might return 2.401E+9 indicating that the frequency corresponding to the X dbm value to the left of the center frequency is 2.401GHz.

READ:OBWidth:BOBW:XDBMbandwidth:IN:RIGHt:FREQuency? (Query Only)

Returns the frequency corresponding to the X dBm value to the right of the center frequency measured in the inward direction.

Conditions Measurement views: Bluetooth 20 dB Bandwith

Group Read commands

Syntax `READ:OBWidth:BOBW:XDBMbandwidth:IN:RIGHt:FREQuency?`

Related Commands READ:OBWidth:BOBW:XDBMbandwidth:IN:LEFT:FREQuency?

Arguments None

Returns <value> ::= <NRf> the right frequency in Hz.

Examples `READ:OBWIDTH:BOBW:XDBMBANDWIDTH:IN:RIGHT:FREQUENCY?` might return 2.445E+9 indicating that the frequency corresponding to the XdBm value to the right of the center the frequency is 2.445 GHz.

READ:OBWidth:BOBW:XDBMbandwidth:OUT:BANDwidth? (Query Only)

Returns the difference between the higher and lower frequency points corresponding to the value which it is the X dBm value measured outwards (from the center frequency to the ends).

Conditions Measurement views: Bluetooth 20 dB Bandwith

Group Read commands

Syntax `READ:OBWidth:BOBW:XDBMbandwidth:OUT:BANDwidth?`

Related Commands READ:OBWidth:BOBW:XDBBandwidth?, READ:OBWidth:BOBW:XDBMbandwidth:IN:BANDwidth?

Arguments None

Returns <value> ::= <NRf> the input X dBm value in Hz.

Examples `READ:OBWIDTH:BOBW:XDBMBANDWIDTH:OUT:BANDWIDTH?` might return 1.2518400000E+6 indicating that the bandwidth corresponding to the output XdBm vaue is 1.2518 MHz measured in the outward direction.

READ:OBWidth:BOBW:XDBMbandwidth:OUT:LEFT:FREQuency? (Query Only)

Returns the frequency corresponding to the X dBm value to the left of the center frequency measured in the outward direction.

Conditions Measurement views: Bluetooth 20 dB Bandwith

Group Read commands

Syntax `READ:OBWidth:BOBW:XDBMbandwidth:OUT:LEFT:FREQuency?`

Related Commands READ:OBWidth:BOBW:XDBMbandwidth:OUT:RIGHt:FREQuency?

Arguments None

Returns <value> ::= <NRf> the left frequency in Hz.

Examples `READ:OBWIDTH:BOBW:XDBMBANDWIDTH:OUT:LEFT:FREQUENCY?` might return 2.401E+9 indicating that the frequency corresponding to the X dbm value to the left of the center frequency is 2.401GHz.

READ:OBWidth:BOBW:XDBMbandwidth:OUT:RIGHt:FREQuency? (Query Only)

Returns the frequency corresponding to the X dBm value to the right of the center frequency measured in the outward direction.

Conditions Measurement views: Bluetooth 20 dB Bandwith

Group Read commands

Syntax `READ:OBWidth:BOBW:XDBMbandwidth:OUT:RIGHt:FREQuency?`

Related Commands READ:OBWidth:BOBW:XDBMbandwidth:OUT:LEFT:FREQuency?

Arguments None

Returns <value> ::= <NRf> the Right frequency in Hz.

Examples `READ:OBWIDTH:BOBW:XDBMBANDWIDTH:OUT:RIGHT:FREQUENCY?` might return 2.445E+9 indicating that the frequency corresponding to the XdBm value to the right of the center the frequency is 2.445 GHz

READ:OBWidth:FREQuency:ERRor? (Query Only)

Returns the frequency error in the Occupied Bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Read commands

Syntax `READ:OBWidth:FREQuency:ERRor?`

Arguments None

Returns `<freq_error> ::` `<NRf>` is the frequency error in Hz.

Examples `READ:OBWIDTH:FREQUENCY:ERROR?` might return `-10.7E+3`, indicating the frequency error is -10.7 kHz.

READ:OBWidth:OBWidth:BANDwidth? (Query Only)

Returns the occupied bandwidth in the Occupied Bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Read commands

Syntax `READ:OBWidth:OBWidth:BANDwidth?`

Arguments None

Returns `<OBW> ::` `<NRf>` is the occupied bandwidth in Hz.

Examples `READ:OBWIDTH:OBWIDTH:BANDWIDTH?` might return `4.0E+6`, indicating the occupied bandwidth is 4 MHz.

READ:OBWidth:OBWidth:LEFT:FREQuency? (Query Only)

Returns the left (lower) frequency of the occupied bandwidth.

Conditions Measurement views: Occupied Bandwidth

Group Read commands

Syntax `READ:OBWidth:OBWidth:LEFT:FREQuency?`

Related Commands READ:OBWidth:OBWidth:RIGHt:FREQuency?

Arguments None

Returns `<OBW_left_freq> ::` `<NRf>` is the left frequency in Hz.

Examples `READ:OBWIDTH:OBWIDTH:LEFT:FREQUENCY?` might return `1.498E+9`, indicating the left frequency is 1.498 GHz.

READ:OBWidth:OBWidth:LEFT:LEVel? (Query Only)

Returns the level at the left frequency of the occupied bandwidth.

Conditions Measurement views: Occupied Bandwidth

Group Read commands

Syntax `READ:OBWidth:OBWidth:LEFT:LEVel?`

Related Commands READ:OBWidth:OBWidth:RIGHt:LEVel?

Arguments None

Returns `<OBW_left_level> ::` `<NRf>` is the level at the left frequency in dB.

Examples `READ:OBWIDTH:OBWIDTH:LEFT:LEVEL?` might return `-23.5`, indicating the level at the left frequency is -23.5 dB.

READ:OBWidth:OBWidth:POWer? (Query Only)

Returns the reference power in the Occupied Bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Read commands

Syntax `READ:OBWidth:OBWidth:POWer?`

Arguments None

Returns `<OBW_ref_power> ::` `<NRf>` is the reference power in dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `READ:OBWIDTH:OBWIDTH:POWER?` might return `-10.0`, indicating the reference power is -10 dBm.

READ:OBWidth:OBWidth:RIGHt:FREQuency? (Query Only)

Returns the right (higher) frequency of the occupied bandwidth.

Conditions Measurement views: Occupied Bandwidth

Group Read commands

Syntax `READ:OBWidth:OBWidth:RIGHt:FREQuency?`

Related Commands READ:OBWidth:OBWidth:LEFT:FREQuency?

Arguments None

Returns `<OBW_right_freq> ::` `<NRf>` is the right frequency in Hz.

Examples `READ:OBWIDTH:OBWIDTH:RIGHT:FREQUENCY?` might return `1.502E+9`, indicating the right frequency is 1.502 GHz.

READ:OBWidth:OBWidth:RIGHt:LEVel? (Query Only)

Returns the level at the right frequency of the occupied bandwidth.

Conditions Measurement views: Occupied Bandwidth

Group Read commands

Syntax `READ:OBWidth:OBWidth:RIGHt:LEVel?`

Related Commands READ:OBWidth:OBWidth:LEFT:LEVel?

Arguments None

Returns <OBW_right_level> :: <NRf> is the level at the right frequency in dB.

Examples READ:OBWIDTH:OBWIDTH:RIGHT:LEVEL? might return -23.5, indicating the level at the right frequency is -23.5 dB.

READ:OBWidth:SPECtrum? (Query Only)

Returns spectrum trace data of the Occupied Bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Read commands

Syntax READ:OBWidth:SPECtrum?

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the amplitude in dBm at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples READ:OBWIDTH:SPECTRUM? might return #43204xxxx... (3204-byte data) for the spectrum trace data of the Occupied Bandwidth measurement.

READ:OBWidth:XDBBandwidth:BANDwidth? (Query Only)

Returns the x dB bandwidth in the Occupied Bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Read commands

Syntax `READ:OBWidth:XDBBandwidth:BANDwidth?`

Arguments None

Returns `<xdBBW> ::` `<NRf>` is the x dB bandwidth in Hz.

Examples `READ:OBWIDTH:XDBBANDWIDTH:BANDWIDTH?` might return `2.0E+6`, indicating the x dB bandwidth is 2 MHz.

READ:OBWidth:XDBBandwidth:LEFT:FREQuency? (Query Only)

Returns the left (lower) frequency of the x dB bandwidth.

Conditions Measurement views: Occupied Bandwidth

Group Read commands

Syntax `READ:OBWidth:XDBBandwidth:LEFT:FREQuency?`

Related Commands READ:OBWidth:XDBBandwidth:RIGHt:FREQuency?

Arguments None

Returns `<xdBBW_left_freq> ::` `<NRf>` is the left frequency in Hz.

Examples `READ:OBWIDTH:XDBBANDWIDTH:LEFT:FREQUENCY?` might return `1.498E+9`, indicating the left frequency is 1.498 GHz.

READ:OBWidth:XDBBandwidth:LEFT:LEVel? (Query Only)

Returns the level at the left frequency of the x dB bandwidth.

Conditions Measurement views: Occupied Bandwidth

Group Read commands

Syntax `READ:OBWidth:XDBBandwidth:LEFT:LEVel?`

Related Commands READ:OBWidth:XDBBandwidth:RIGHt:LEVel?

Arguments None

Returns `<xdBBW_left_level> ::` `<NRf>` is the level at the left frequency in dB.

Examples `READ:OBWIDTH:XDBBANDWIDTH:LEFT:LEVEL?` might return `-23.5`, indicating the level at the left frequency is -23.5 dB.

READ:OBWidth:XDBBandwidth:POWer? (Query Only)

Returns the reference power in the x dB bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Read commands

Syntax `READ:OBWidth:XDBBandwidth:POWer?`

Arguments None

Returns `<xdBBW_ref_power> ::` `<NRf>` is the reference power in dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `READ:OBWIDTH:XDBBANDWIDTH:POWER?` might return `-10.0`, indicating the reference power is -10 dBm.

READ:OBWidth:XDBBandwidth:RIGHt:FREQuency? (Query Only)

Returns the right (higher) frequency of the x dB bandwidth.

Conditions Measurement views: Occupied Bandwidth

Group Read commands

Syntax `READ:OBWidth:XDBBandwidth:RIGHt:FREQuency?`

Related Commands READ:OBWidth:XDBBandwidth:LEFT:FREQuency?

Arguments None

Returns `<xdBBW_right_freq> ::` `<NRf>` is the right frequency in Hz.

Examples `READ:OBWIDTH:XDBBANDWIDTH:RIGHT:FREQUENCY?` might return `1.502E+9`, indicating the right frequency is 1.502 GHz.

READ:OBWidth:XDBBandwidth:RIGHt:LEVel? (Query Only)

Returns the level at the right frequency of the x dB bandwidth.

Conditions Measurement views: Occupied Bandwidth

Group Read commands

Syntax `READ:OBWidth:XDBBandwidth:RIGHt:LEVel?`

Related Commands READ:OBWidth:XDBBandwidth:LEFT:LEVel?

Arguments None

Returns `<xdBBW_right_level> ::` `<NRf>` is the level at the right frequency in dB.

Examples `READ:OBWIDTH:XDBBANDWIDTH:RIGHT:LEVEL?` might return `-23.5`, indicating the level at the right frequency is -23.5 dB.

READ:OFDM:APOWer? (Query Only)

Returns the average power in the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:APOWer?`

Arguments None

Returns <apower>::<NRf> the average power in dB.

Examples `READ:OFDM:APOWER?` might return `–23.4584459235` indicating the average power is –23.46 dB.

READ:OFDM:APOWer:PEAK? (Query Only)

Returns the peak-to-average power in the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:APOWer:PEAK?`

Arguments None

Returns <apower_peak>::<NRf> is the peak-to-average power in dBm.

Examples `READ:OFDM:APOWER:PEAK?` might return `10.4140096289` indicating that peak-to-average power is 10.41 dBm.

READ:OFDM:CONSte:MAGNitude? (Query Only)

Returns the constellation magnitude data for the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:CONSte:MAGNitude?`

Related Commands READ:OFDM:CONSte:PHASe?
READ:OFDM:CONSte:TYPE?
READ:OFDM:CONSte:VALue?

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the data value at the n^{th} data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:OFDM:CONSTE:MAGNITUDE?` might return `#510400xxxx...` (10400-byte data) for the measurement.

READ:OFDM:CONSte:PHASe? (Query Only)

Returns the constellation phase data for the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:CONSte:PHASe?`

Related Commands READ:OFDM:CONSte:MAGNitude?
READ:OFDM:CONSte:TYPE?
READ:OFDM:CONSte:VALue?

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the data value at the nth data point in 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:OFDM:CONSTE:PHASE?` might return `#510400xxxx...` (10400-byte data) for the measurement.

READ:OFDM:CONSte:TYPE? (Query Only)

Returns the constellation context value of the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:CONSte:TYPE?`

Related Commands READ:OFDM:CONSte:MAGNitude?
READ:OFDM:CONSte:PHASe?
READ:OFDM:CONSte:VALue?

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the data value at the nth data point in 16-bit little endian integers. The integers must be translated to the proper context as listed below.

Integer value	Type (context)
0	Pilot
1	Data
2	Unused or Null

Examples `READ:OFDM:CONSTE:TYPE?` might return `#41352` (1352-byte data) for the measurement. The actual data must be decoded to the context type using the table above.

READ:OFDM:CONSte:VALue? (Query Only)

Returns the constellation value of the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:CONSte:VALue?`

Related Commands READ:OFDM:CONSte:MAGNitude?
READ:OFDM:CONSte:PHASe?
READ:OFDM:CONSte:TYPE?

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the data value at the n^{th} data point in 16-bit little endian integers.

Examples `READ:OFDM:CONSTE:VALUE?` might return `#41352` indicating 1352 bytes of data for the measurement.

READ:OFDM:CPE? (Query Only)

Returns the Common Pilot Error magnitude for the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:CPE?`

Arguments None

Returns <NRf> the RMS magnitude error in percent.

Examples `READ:OFDM:CPE?` might return `3.7868041505` indicating that the CPE is 3.787%.

READ:OFDM:CRESponse:MAGNitude? (Query Only)

Returns the channel response magnitude data for the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:CRESponse:MAGNitude?`

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the data value at the n^{th} data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:OFDM:CRESPONSE:MAGNITUDE?` might return `#3804xxxx. . .` (804 byte data) for the measurement.

READ:OFDM:CRESponse:PHASe? (Query Only)

Returns the channel response phase data for the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:CRESponse:PHASe?`

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the data value at the n^{th} data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:OFDM:CRESPONSE:PHASE?` might return `#3804xxxx. . .` (804 byte data) for the measurement.

READ:OFDM:EVM:PEAK:DECibel:ALL? (Query Only)

Returns the peak EVM data for all subcarriers in the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:EVM:PEAK:DECibel:ALL?`

Arguments None

Returns <NRf> data in dB.

Examples `READ:OFDM:EVM:PEAK:DECIBEL:ALL?` might return `-20.1872549032` indicating the data is –20.19 dB for the measurement.

READ:OFDM:EVM:PEAK:DECibel:DATA? (Query Only)

Returns the peak EVM data for the data subcarriers in the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:EVM:PEAK:DECibel:DATA?`

Arguments None

Returns <NRf> data in dB.

Examples `READ:OFDM:EVM:PEAK:DECIBEL:DATA?` might return `-20.1872549032` indicating the data is –20.19 dB for the measurement.

READ:OFDM:EVM:PEAK:DECibel:PILots? (Query Only)

Returns the peak EVM data for the pilot subcarriers in the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:EVM:PEAK:DECibel:PILots?`

Arguments None

Returns <NRf> data in dB.

Examples `READ:OFDM:EVM:PEAK:DECIBEL:PILOTS?` might return `-20.2872549032` indicating the data is –20.29 dB for the measurement.

READ:OFDM:EVM:PEAK:PERCent:ALL? (Query Only)

Returns the peak EVM data for all subcarriers in the OFDM measurement as a percent.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:EVM:PEAK:PERCent:ALL?`

Arguments None

Returns <NRf> data in percent.

Examples `READ:OFDM:EVM:PEAK:PERCENT:ALL?` might return `19.3223863840` indicating the data is 19.322% for the measurement.

READ:OFDM:EVM:PEAK:PERCent:DATA? (Query Only)

Returns the peak EVM data for the data subcarriers in the OFDM measurement as a percent.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:EVM:PEAK:PERCent:DATA?`

Arguments None

Returns <NRf> data in percent.

Examples `READ:OFDM:EVM:PEAK:PERCENT:DATA?` might return `19.3223863840` indicating the data is 19.322% for the measurement.

READ:OFDM:EVM:PEAK:PERCent:PILots? (Query Only)

Returns the peak EVM data for the pilot subcarriers in the OFDM measurement as a percent.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:EVM:PEAK:PERCent:PILots?`

Arguments None

Returns <NRf> data in percent.

Examples `READ:OFDM:EVM:PEAK:PERCENT:PILOTS?` might return `10.8543863840` indicating the data is 10.854% for the measurement.

READ:OFDM:EVM:PEAK:SCARrier:ALL? (Query Only)

Returns the peak EVM data for all subcarriers at the subcarrier level in the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:EVM:PEAK:SCARrier:ALL?`

Arguments None

Returns <NR1>

Examples `READ:OFDM:EVM:PEAK:SCARRIER:ALL?` might return `99.00000` indicating the data is 99 for the measurement.

READ:OFDM:EVM:PEAK:SCARrier:DATA? (Query Only)

Returns the peak EVM data for the data subcarriers at the subcarrrier level in the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:EVM:PEAK:SCARrier:DATA?`

Arguments None

Returns <NR1>

Examples `READ:OFDM:EVM:PEAK:SCARRIER:DATA?` might return `99.00000` indicating the data is 99 for the measurement.

READ:OFDM:EVM:PEAK:SCARrier:PILots? (Query Only)

Returns the peak EVM data for the pilot subcarriers at the subxcarrier level in the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:EVM:PEAK:SCARrier:PILots?`

Arguments None

Returns <NR1>

Examples `READ:OFDM:EVM:PEAK:SCARRIER:PILOTS?` might return `88.00000` indicating the data is 88for the measurement

READ:OFDM:EVM:PEAK:SYMBol:ALL? (Query Only)

Returns the peak EVM data for all subcarriers at the symbol level in the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:EVM:PEAK:SYMBol:ALL?`

Arguments None

Returns <NR1>

Examples `READ:OFDM:EVM:PEAK:SYMBOL:ALL?` might return `2.00000` indicating the data is 2 for the measurement

READ:OFDM:EVM:PEAK:SYMBol:DATA? (Query Only)

Returns the peak EVM data for the data subcarriers at the symbol level in the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:EVM:PEAK:SYMBol:DATA?`

Arguments <NR1>

Returns

Examples `READ:OFDM:EVM:PEAK:SYMBOL:DATA?` might return `2.00000` indicating the data is 2 for the measurement

READ:OFDM:EVM:PEAK:SYMBol:PILots? (Query Only)

Returns the peak EVM data for the pilot subcarriers at the symbol level in the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:EVM:PEAK:SYMBol:PILots?`

Arguments <NR1>

Returns

Examples `READ:OFDM:EVM:PEAK:SYMBOL:PILOTS?` might return `1.00000` indicating the data is 1for the measurement

READ:OFDM:EVM:RMS:DECibel:ALL? (Query Only)

Returns the RMS EVM data for all subcarriers in the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:EVM:RMS:DECibel:ALL?`

Arguments None

Returns <NRf> data in dB.

Examples `READ:OFDM:EVM:RMS:DECIBEL:ALL?` might return `–26.9012093267` indicating the data is –26.90 for the measurement.

READ:OFDM:EVM:RMS:DECibel:DATA? (Query Only)

Returns the RMS EVM data for the data subcarriers in the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:EVM:RMS:DECibel:DATA?`

Arguments None

Returns <NRf> data in dB.

Examples `READ:OFDM:EVM:RMS:DECIBEL:DATA?` might return `-26.8477116269` indicating the data is –26.85 for the measurement.

READ:OFDM:EVM:RMS:DECibel:PILots? (Query Only)

Returns the RMS EVM data for the pilot subcarriers in the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:EVM:RMS:DECibel:PILots?`

Arguments None

Returns <NRf> data in dB.

Examples `READ:OFDM:EVM:RMS:DECIBEL:PILOTS?` might return `-33.0589143032` indicating the data is –33.06 dB.

READ:OFDM:EVM:RMS:PERCent:ALL? (Query Only)

Returns the peak RMS data for all subcarriers in the OFDM measurement as a percent.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:EVM:RMS:PERCent:ALL?`

Arguments None

Returns <NRf> data in percent.

Examples `READ:OFDM:EVM:RMS:PERCENT:ALL?` might return `225.0743627548` indicating the data is 225.074% for the measurement.

READ:OFDM:EVM:RMS:PERCent:DATA? (Query Only)

Returns the peak RMS data for the data subcarriers in the OFDM measurement as a percent.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:EVM:RMS:PERCent:DATA?`

Arguments None

Returns <NRf> data in percent.

Examples `READ:OFDM:EVM:RMS:PERCENT:DATA?` might return `260.6973409653` indicating that the data is 260.697% for the measurement.

READ:OFDM:EVM:RMS:PERCent:PILots? (Query Only)

Returns the peak RMS data for the pilot subcarriers in the OFDM measurement as a percent.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:EVM:RMS:PERCent:PILots?`

Arguments None

Returns <NRf> data in percent.

Examples `READ:OFDM:EVM:RMS:PERCENT:PILOTS?` might return `210.8103863840` indicating the data is 210.810% for the measurement.

READ:OFDM:EVM:TRACe<x>? (Query Only)

Returns the EVM trace data for the OFDM measurement.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbols.

When <x> is 3, the parameter is Average versus Subcarrier.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:EVM:TRACe<x>?`

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the data value at the n^{th} data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:OFDM:EVM:TRACE2?` might return `#252xxxx. . .` (52 byte data) for the Average versus Symbols measurement.

READ:OFDM:FERRor? (Query Only)

Returns the Frequency error reading for the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:FERRor?`

Arguments None

Returns <NRf> frequency error.

Examples `READ:OFDM:FERROR?` might return `82.8617142098E+3` indicating the frequency error was 82.86 kHz.

READ:OFDM:FLATness:PASS? (Query Only)

Queries whether the average power level values of subcarriers across the signal bandwidth on the OFDM Spectral Flatness display remain within the limits defined for a particular standard.

Conditions Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group Read commands

Syntax `READ:OFDM:FLATness:PASS?`

Returns Returns "1" in the case that all data remains within the defined limits, and "0" otherwise.

Examples `READ:OFDM:FLAT:PASS?` might return 1, which means that the average power level values of subcarriers across the signal bandwidth remain within the defined limits.

READ:OFDM:FLATness:RESult? (Query Only)

Returns the summary results of each segment of the OFDM Spectral Flatness display.

Conditions Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group Read commands

Syntax `READ:OFDM:FLATness:RESult?`

Returns The data is returned as 12 comma delineated text values. The specific values are :

- minimum subcarrier within segment 1
- minimum subcarrier average energy deviation from the global average within segment 1
- deviation the of minimum subcarrier average from the allowed value in segment 1
- maximum subcarrier within segment 1
- maximum subcarrier average energy deviation from the global average within segment 1
- deviation of the maximum subcarrier average from the allowed value in segment 1
- minimum subcarrier within segment 2
- minimum subcarrier average energy deviation from the global average within segment 2
- deviation the of minimum subcarrier average from the allowed value in segment 2
- maximum subcarrier within segment 2
- maximum subcarrier average energy deviation from the global average within segment 2
- deviation of the maximum subcarrier average from the allowed value in segment 2

Examples `READ:OFDM:FLAT:RES?` might return might return `-6.0000000000,-17.0626174659E-3,3.9829373825,12.0000000000,19.979260` which represents the summary results of each segment of the OFDM Spectral Flatness display.

READ:OFDM:FLATness:TRACe<x>? (Query Only)

Returns the OFDM Spectral Flatness trace data. When <x> is 1, the parameter is Matrix (symbol deviation per subcarrier). When<x> is 2, the parameter is Average Deviation vs Subcarrier.

Conditions Measurement view: OFDM

This command requires Option 22, "OFDM Measurements".

Group Read commands

Syntax `READ:OFDM:FLATness:TRACe<x>?`

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where:

<num_digit> is the number of digits in <num_byte.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:OFDM:FLAT:TRACE2?` might return `#510400..` (10400-byte data) which represents the data for the Average Deviation vs. Subcarrier trace.

READ:OFDM:GAIN:IMBalance? (Query Only)

Returns the gain imbalance for the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:GAIN:IMBalance?`

Arguments None

Returns <NRf> gain imbalance in dB.

Examples `READ:OFDM:GAIN:IMBALANCE?` might return `-57.746E-3` indicating that the gain imbalance is –0.057746 dB.

READ:OFDM:IQ:ORIGin:OFFSet? (Query Only)

Returns the IQ origin offset for the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:IQ:ORIGin:OFFSet?`

Arguments None

Returns <NRf> gain imbalance in dB.

Examples `READ:OFDM:IQ:ORIGIN:OFFSET?` might return `-53.47017` indicating that the IQ origin offset is –53.47 dB.

READ:OFDM:MERRor:TRACe<x>? (Query Only)

Returns the magnitude error trace data for the OFDM measurement.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbols.

When <x> is 3, the parameter is Average versus Subcarrier.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:MERRor:TRACe<x>?`

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the data value at the n^{th} data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:OFDM:MERROR:TRACE2` might return `#3108xxxx. . .` (108-byte data) for the average versus symbols trace measurement.

READ:OFDM:PACKet:DIRection? (Query Only)

Returns the direction of the packet in the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:PACKet:DIRection?`

Arguments None

Returns One of the following values.

`UP` specifies that the packet is in the up direction.

`DOWN` specifies that the packet is in the down direction.

`BOTH` specifies that the packet is both directions.

Examples `READ:OFDM:PACKET:DIRECTION?` might return `DOWN` indicating that the packet was in the down direction.

READ:OFDM:PERRor:TRACe<x>? (Query Only)

Returns the phase error trace data for the OFDM measurement.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbols.

When <x> is 3, the parameter is Average versus Subcarrier.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:PERRor:TRACe<x>?`

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the data value at the n^{th} data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:OFDM:PERROR:TRACE2` might return `#3100xxxx. . .` (100-byte data) for the average versus symbols trace measurement.

READ:OFDM:POWer:TRACe<x>? (Query Only)

Returns the power trace data for the OFDM measurement.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbols.

When <x> is 3, the parameter is Average versus Subcarrier.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:POWer:TRACe<x>?`

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the data value at the n^{th} data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:OFDM:POWER:TRACE2` might return `#296xxxx. . .` (96-byte data) for the average versus symbols trace measurement.

READ:OFDM:QUADrature:OFFSet? (Query Only)

Returns the quadrature offset in the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:QUADrature:OFFSet?`

Arguments None

Returns <NRf> quadrature offset in degrees.

Examples `READ:OFDM:QUADRATURE:OFFSET?` might return `–99.9` indicating that the offset is –99°.

READ:OFDM:SCARriers? (Query Only)

Returns the number of subcarriers in the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:SCARriers?`

Arguments None

Returns <NR1> number of subcarriers.

Examples `READ:OFDM:SCARRIERS?` might return `200.0000` indicating that there are 200 subcarriers in the measurement.

READ:OFDM:SCARriers:SPACing? (Query Only)

Returns the subcarrier spacing for the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:SCARriers:SPACing?`

Arguments None

Returns <NRf> subcarrier spacing.

Examples `READ:OFDM:SCARRIERS:SPACING?` might return `90.000E+3` indicating that the subcarrier spacing is 90.000 kHz.

READ:OFDM:STABle:VALUe? (Query Only)

Returns the number of bytes in the symbol table for the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:STABle:VALUe?`

Arguments None

Returns #<num_digit><num_byte>

Where

<num_digit> is the number of digits in <num_byte>

<num_byte> is the number of bytes of data.

Examples `READ:OFDM:STABLE:VALUE?` might return `#43848` indicating that there are 3848 bytes in the symbol table.

READ:OFDM:SYMBol:CERRor? (Query Only)

Returns the symbol clock error for the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:SYMBol:CERRor?`

Arguments None

Returns <NRf> symbol clock error in ppm.

Examples `READ:OFDM:SYMBOL:CERROR?` might return `422.7135479929` indicating that the symbol clock error was 422.714 ppm.

READ:OFDM:SYMBol:COUNt? (Query Only)

Returns the number of symbols for the OFDM measurement.

Conditions Measurement views: OFDM

Group Read commands

Syntax `READ:OFDM:SYMBol:COUNt?`

Arguments None

Returns <NR1> number of symbols in the measurement.

Examples `READ:OFDM:SYMBOL:COUNT?` might return `25.00000` indicating that there were 25 symbols for the measurement.

READ:P25:CONSte:FERRor? (Query Only)

Returns the frequency error in Hz. The frequency error is the difference between the measured carrier frequency of the signal and the user-selected center frequency of the analyzer.

When Frequency Error is in Auto mode, this query returns the frequency error value. When in Manual mode, it returns the frequency offset.

Conditions Measurement view: P25 Constellation

Group Read commands

Syntax `READ:P25:CONSte:FERRor?`

Returns <freq_error> :: <NRf> is the frequency error in Hz.

Examples `READ:P25:CONSTE:FERROR?` might return -10.7E+3, which is a frequency error of -10.7 kHz.

READ:P25:EDIagram:FDEViation? (Query Only)

Returns the Frequency Deviation vs Time trace data with the X values.

Conditions Measurement view: P25 Eye Diagram

Group Read commands

Syntax `READ:P25:EDIagram:FDEViation?`

Returns #<num_digit><num_byte><Y(1)><X(1)><Y(2)><X(2)>...<Y(n)><X(n)>

Where:

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<Y(n)> and <X(n)> are the frequency deviation in Hz and time (symbols) coordinate pair at the *n*th data point. 4-byte little endian floating point format specified in IEEE 488.2.

Examples `READ:P25:EDIAGRAM:FDEVIATION?` might return #3160xxxx... (160-byte data) For the Frequency Deviation vs Time trace.

READ:P25:EDIagram:FERRor? (Query Only)

Returns the frequency error in Hz. The frequency error is the difference between the measured carrier frequency of the signal and the user-selected center frequency of the analyzer.

Conditions Measurement view: P25 Eye Diagram

Group Read commands

Syntax `READ:P25:EDIagram:FERRor?`

Returns <freq_error> :: <NRf> is the frequency error in Hz.

Examples `READ:P25:EDIAGRAM:FERROR?` might return -10.7E+3, which is a frequency error of -10.7 kHz.

READ:P25:PVTime:BURSt:POWer? (Query Only)

Returns the average power of the burst packet, in dBm, for the P25 Power vs. Time trace.

Conditions Measurement view: P25 Power vs. Time

Group Read commands

Syntax `READ:P25:PVTime:BURSt:POWer?`

Returns Floating point number that represents the average power of the burst packet, in dBm.

Examples `READ:P25:PVTIME:BURST:POWER?` might return -1.99, which represents the value of the average power of the burst packet of -1.99 dBm.

READ:P25:PVTime:FALL:TIMe? (Query Only)

Queries the value of the width of the falling edge, in seconds.

Conditions Measurement view: P25 Power vs. Time

Group Read commands

Syntax `READ:P25:PVTime:FALL:TIMe?`

Related Commands READ:P25:PVTime:FTIMe?

Returns Floating point number which represents the value of the width of the falling edge of the burst, in seconds.

Examples `READ:P25:PVTIME:FALL:TIME?` might return 7e-6, which indicates the width of the falling edge is 7 µs.

READ:P25:PVTime:FTIMe? (Query Only)

Queries the value of the width of the falling edge, in seconds.

Conditions Measurement view: P25 Power vs. Time

Group Read commands

Syntax `READ:P25:PVTime:FTIMe?`

Related Commands READ:P25:PVTime:FALL:TIMe?

Returns Floating point number which represents the value of the width of the falling edge of the burst, in seconds.

Examples `READ:P25:PVTIME:FTIME?` might return 7e-6, which indicates the width of the falling edge is 7 µs.

READ:P25:PVTime:TRACe:X? (Query Only)

Returns the horizontal values (time in seconds) for the P25 Power vs. Time trace.

Conditions Measurement view: P25 Power vs. Time

Group Read commands

Syntax `READ:P25:PVTime:TRACe:X?`

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where:

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the *n*th data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:P25:PVTIME:TRACE:X?` might return #43204 (3204-byte data), which represents the value of the time in seconds (horizontal values) for the P25 Power vs. Time trace.

READ:P25:PVTime:TRACe:XY? (Query Only)

Returns the horizontal value (time in seconds) and vertical value (power) for the P25 Power vs. Time trace <x(n)><y(n)> is the horizontal value and vertical value pair at the *n*th data point.

Conditions Measurement view: P25 Power vs. Time

Group Read commands

Syntax `READ:P25:PVTime:TRACe:XY?`

Related Commands READ:P25:PVTime:TRACe:X?

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where:

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the *n*th data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:P25:PVTIME:TRACE:XY?` might return #43204 (3204-byte data), which represents the horizontal value (time in seconds) and vertical value (power) pair at the *n*th data point.

READ:P25:SUMMary:MODUlation:MEASurement:FIDelity? (Query Only)

Returns the Modulation Fidelity measurement results.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax `READ:P25:SUMMary:MODUlation:MEASurement:FIDelity?`

Returns Modulation Fidelity measurement result

Examples `READ:P25:SUMMARY:MODULATION:MEASUREMENT:FIDELITY?`

READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:ACCUracy? (Query Only)

Returns the Operational Frequency Accuracy measurement results.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax `READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:ACCUracy?`

Returns Frequency Accuracy measurement result

Examples `READ:P25:SUMMARY:MODULATION:MEASUREMENT:FREQUENCY:ACCURACY?`

READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:NEGPeak

Returns the Frequency Deviation, negative peak, measurement results.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax `READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:NEGPeak`

Arguments <integer,integer>

Returns Frequency Deviation, negative peak, result

Examples `READ:P25:SUMMARY:MODULATION:MEASUREMENT:FREQUENCY:DEVIATION:NEGPEAK<2,6>` indicates the coordinate value of the Frequency Deviation negative peak.

READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:POSPeak

Returns the Frequency Deviation, positive peak, measurement results.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax `READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:DEVIation:POSPeak`

Arguments <integer,integer>

Returns Frequency Deviation, positive peak, result

Examples `READ:P25:SUMMARY:MODULATION:MEASUREMENT:FREQUENCY:DEVIATION:POSPEAK<-2,-6>` indicates the coordinate value of the Frequency Deviation positive peak.

READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:ERRor

Returns the Frequency Error measurement results.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax `READ:P25:SUMMary:MODUlation:MEASurement:FREQuency:ERRor`

Arguments None

Returns Frequency Error measurement result

Examples `READ:P25:SUMMARY:MODULATION:MEASUREMENT:FREQUENCY:ERROR?`

READ:P25:SUMMary:MODUlation:MEASurement:SRA

Returns the Symbol Rate Accuracy measurement results.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax `READ:P25:SUMMary:MODUlation:MEASurement:SRA`

Arguments None

Returns Symbol Rate Accuracy measurement result

Examples
```
READ:P25:SUMMARY:MODULATION:MEASUREMENT:SRA?
```

READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF? (Query Only)

This command returns the HCPM Pmax-off measurement results.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax
```
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXOFF?
```

Returns HCPM Pmax-off measurement results

Examples
```
READ:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MAXOFF?
```

READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON? (Query Only)

This command returns the HCPM Pmax-on measurement results.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax
```
READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXON?
```

Returns HCPM Pmax-on measurement results

Examples
```
READ:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MAXON?
```

READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSs? (Query Only)

Returns the HCPM Pss-max measurement results.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax `READ:P25:SUMMary:POWEr:MEASurement:HCPM:MAXSs?`

Returns HCPM Pss-on measurement results

Examples `READ:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MAXSS?`

READ:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum? (Query Only)

Returns the HCPM Pss-min measurement results.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax `READ:P25:SUMMary:POWEr:MEASurement:HCPM:MINImum?`

Returns HCPM Pss-min measurement results

Examples `READ:P25:SUMMARY:POWER:MEASUREMENT:HCPM:MINIMUM?`

READ:P25:SUMMary:POWEr:MEASurement:HCPM:OFFSlot? (Query Only)

Returns the HCPM Offslot power results.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax `READ:P25:SUMMary:POWEr:MEASurement:HCPM:OFFSlot?`

Returns HCPM Offslot power results

Examples `READ:P25:SUMMARY:POWER:MEASUREMENT:HCPM:OFFSLOT?`

READ:P25:SUMMary:POWEr:MEASurement:RF? (Query Only)

Returns the RF output power results.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax `READ:P25:SUMMary:POWEr:MEASurement:RF?`

Returns RF output power result

Examples `READ:P25:SUMMARY:POWER:MEASUREMENT:RF?`

READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE? (Query Only)

Returns the HCPM Time alignment t_error_1 measurement results.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax `READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE?`

Returns HCPM Time alignment t_error_1 measurement results.

Examples `READ:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:TERROR:ONE?`

READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ONE:ACQ (Query Only)

Returns the HCPM Time alignment's t_error_1 acquisition count over which the t_error_1 averaging is done.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax
```
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:
TERRor:ONE:ACQ:COUNT?
```

Returns HCPM Time alignment's t_error_1 acquisition count over which the t_error_1 averaging is done.

Examples
```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:
TERROR:ONE:ACQ:COUNT?
```

READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO? (Query Only)

Returns the HCPM Time alignment's t_error_0 measurement results over which the t_error_0 averaging is done.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax
```
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:
TERRor:ZERO?
```

Returns HCPM Time alignment's t_error_0 measurement results over which the t_error_0 averaging is done.

Examples
```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:
TERROR:ZERO?
```

READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TERRor:ZERO:ACQ:COUNt? (Query Only)

Returns the HCPM Time alignment's t_error_0 acquisition count over which the t_error_0 averaging is done.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax
```
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:
TERRor:ZERO:ACQ:COUNT?
```

Returns HCPM Time alignment's t_error_0 acquisition count.

Examples
```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:
TERROR:ZERO:ACQ:COUNT?
```

READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TOBSync? (Query Only)

Returns the HCPM Time alignment's t_obsync measurement results.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax
```
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:
TOBSync?
```

Returns HCPM Time alignment's t_obsync measurement results.

Examples
```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:
TOBSYNC?
```

READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:TOBSync:ACQ:CO (Query Only)

Returns the HCPM Time alignment's t_obsync acquisition counts over which the t_obsync averaging is done.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax
```
READ:P25:SUMMary:TRIGger:MEASurement:HCPM:TIME:ALIGNment:
TOBSync:ACQ:COUNT?
```

Returns HCPM Time alignment's t_obsync acquisition counts.

Examples
```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:HCPM:TIME:ALIGNMENT:
TOBSYNC:ACQ:COUNT?
```

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT? (Query Only)

Returns the Average Transmitter Power Attack acquisition count over which the Phase 1 Attack time averaging is done.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax
```
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:
COUNT?
```

Returns Average Transmitter Power Attack acquisition count.

Examples
```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:ACQ:
COUNT?
```

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT:BI? (Query Only)

Returns the Average Transmitter Power Attack time busy/idle acquisition count over which the Phase 1 Attack time averaging is done.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax
```
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ACQ:COUNT:BI?
```

Returns Average Transmitter Power Attack, busy/idle, measurement results.

Examples
```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:ACQ:COUNT:BI?
```

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder? (Query Only)

Returns the Average Transmitter Encoder Attack Time measurement results.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax
```
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder?
```

Returns Average Transmitter Encoder Attack Time measurement results.

Examples
```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:ENCODER?
```

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:ENCOder:BI? (Query Only)

Returns the Average Transmitter Power Encoder Attack Time busy/idle measurement results.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax

```
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:
ENCOder:BI?
```

Returns Average Transmitter Power Encoder Attack Time busy/idle measurement results.

Examples

```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:
ENCODER:BI?
```

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr? (Query Only)

Returns the Average Transmitter Power Attack Time measurement results.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax

```
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:
POWEr?
```

Returns Average Transmitter Power Attack Time measurement results.

Examples

```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:
POWER?
```

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI? (Query Only)

Returns the Average Transmitter Power Attack Time, busy/idle measurement results.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax
```
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:ATTAck:TIME:POWEr:BI?
```

Returns Average Transmitter Power Attack Time, busy/idle, measurement results.

Examples
```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:ATTACK:TIME:POWER:BI?
```

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy? (Query Only)

Returns the Throughput delay measurement results.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax
```
READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy?
```

Returns Throughput delay measurement results.

Examples
```
READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:THRUPUT:DELAY?
```

READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy:ACQ:COUNT? (Query Only)

Returns the Throughput delay acquisition counts over which the Throughput delay averaging is done.

Conditions Measurement view: P25 Summary

Group Read commands

Syntax `READ:P25:SUMMary:TRIGger:MEASurement:PHASe1:THRUput:DELAy:ACQ:COUNT?`

Returns Throughput delay acquisition counts.

Examples `READ:P25:SUMMARY:TRIGGER:MEASUREMENT:PHASE1:THRUPUT:DELAY:ACQ:COUNT?`

READ:PERRor:FERRor? (Query Only)

Returns the frequency error in the Phase error versus Time measurement.

Conditions Measurement views: Phase error versus Time

Group Read commands

Syntax `READ:PERRor:FERRor?`

Arguments None

Returns `<freq_error> ::` `<NRf>` is the frequency error in Hz.

Examples `READ:PERROR:FERROR?` might return `-10.7E+3`, indicating the frequency error is -10.7 kHz.

READ:PERRor:PEAK? (Query Only)

Returns the peak value in the Phase error versus Time measurement.

Conditions Measurement views: Phase error versus Time

Group Read commands

Syntax `READ:PERRor:PEAK?`

Related Commands READ:PERRor:PINDex?

Arguments None

Returns `<peak> ::` `<NRf>` is the peak phase error in degrees.

Examples `READ:PERROR:PEAK?` might return `0.683`, indicating the peak phase error is 0.683 °.

READ:PERRor:PINDex? (Query Only)

Returns the time at the phase error peak.

Conditions Measurement views: Phase error versus Time

Group Read commands

Syntax `READ:PERRor:PINDex?`

Related Commands READ:PERRor:PEAK?

Arguments None

Returns `<peak_time> ::` `<NRf>` is the time at the phase error peak in symbol number. The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples `READ:PERROR:PINDEX?` might return `68.000`, indicating that the phase error peak is at symbol #68.

READ:PERRor:RMS (Query Only)

Returns the RMS (Root-Mean-Square) value in the Phase error versus Time measurement.

Conditions Measurement views: Phase error versus Time

Group Read commands

Syntax `READ:PERRor:RMS`

Arguments None

Returns `<rms> ::  <NRf>` is the RMS phase error in degrees.

Examples `READ:PERROR:RMS` might return `0.746`, indicating the phase error is 0.746 ° RMS.

READ:PERRor:TRACe? (Query Only)

Returns the Phase error versus Time trace data.

Conditions Measurement views: Phase error versus Time

Group Read commands

Syntax `READ:PERRor:TRACe?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.
<data(n)> is the phase error data for the point n in degrees,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples READ:PERROR:TRACE? might return #42036xxxx... (2036-byte data) for the Phase error versus Time trace.

READ:PHVTime? (Query Only)

Returns the Phase versus Time trace data.

Conditions Measurement views: Phase versus Time

Group Read commands

Syntax READ:PHVTime?

Arguments None

Returns #<num_digit><num_byte><data(1)><data(2)>...<data(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<data(n)> is the phase in degrees at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples READ:PHVTIME? might return #3160xxxx... (160-byte data) for the Phase versus Time trace.

READ:PHVTime:MAXimum? (Query Only)

Returns the maximum value in the Phase versus Time measurement.

Conditions Measurement views: Phase versus Time

Group Read commands

Syntax `READ:PHVTime:MAXimum?`

Related Commands READ:PHVTime:MAXLocation?

Arguments None

Returns `<max> ::` `<NRf>` is the maximum phase in degrees.

Examples `READ:PHVTIME:MAXIMUM?` might return `153.8`, indicating the maximum phase is 153.8 °.

READ:PHVTime:MAXLocation? (Query Only)

Returns the time at which the phase is maximum.

Conditions Measurement views: Phase versus Time

Group Read commands

Syntax `READ:PHVTime:MAXLocation?`

Related Commands READ:PHVTime:MAXimum?

Arguments None

Returns `<max_time> ::` `<NRf>` is the time in seconds at which the phase is maximum.

Examples `READ:PHVTIME:MAXLOCATION?` might return `175.3E-9`, indicating the I or Q level is maximum at 175.3 ns.

READ:PHVTime:MINimum? (Query Only)

Returns the minimum value in the Phase versus Time measurement.

Conditions Measurement views: Phase versus Time

Group Read commands

Syntax `READ:PHVTime:MINimum?`

Related Commands READ:PHVTime:MINLocation?

Arguments None

Returns `<min> ::` `<NRf>` is the minimum phase in degrees.

Examples `READ:PHVTIME:MINIMUM?` might return `-176.3`, indicating the minimum phase is -176.3 °.

READ:PHVTime:MINLocation? (Query Only)

Returns the time at which the phase is minimum.

Conditions Measurement views: Phase versus Time

Group Read commands

Syntax `READ:PHVTime:MINLocation?`

Related Commands READ:PHVTime:MINimum?

Arguments None

Returns `<min_time> ::` `<NRf>` is the time in seconds at which the phase is minimum.

Examples `READ:PHVTIME:MINLOCATION?` might return `450.7E-9`, indicating the phase is minimum at 450.7 ns.

READ:PHVTime:RESult? (Query Only)

Returns the Phase versus Time measurement results.

Conditions Measurement views: Phase versus Time

Group Read commands

Syntax `READ:PHVTime:RESult?`

Arguments None

Returns `<max>,<max_time>,<min>,<min_time>`

Where
`<max> ::` `<NRf>` is the maximum phase in degrees.
`<max_time> ::` `<NRf>` is the time in seconds at which the phase is maximum.
`<min> ::` `<NRf>` is the minimum phase in degrees.
`<min_time> ::` `<NRf>` is the time in seconds at which the phase is minimum.

Examples `READ:PHVTIME:RESULT?` might return `153.8,175.3E-9,-176.3,450.7E-9`, indicating the maximum phase is 153.8 ° at 175.3 ns and the minimum phase is -176.3 ° at 450.7 ns.

READ:PM:PNEGative? (Query Only)

Returns the negative peak phase deviation (–Pk) in the PM measurement.

Conditions Measurement views: PM

Group Read commands

Syntax `READ:PM:PNEGative?`

Arguments None

Returns `<-Pk> ::` `<NRf>` is the negative peak phase deviation in degrees.

Examples `READ:PM:PNEGATIVE?` might return `–23.42`, indicating the positive peak phase deviation is –23.42 °.

READ:PM:PPOSitive? (Query Only)

Returns the positive peak phase deviation (+Pk) in the PM measurement.

Conditions Measurement views: PM

Group Read commands

Syntax `READ:PM:PPOSitive?`

Arguments None

Returns `<+Pk> ::` `<NRf>` is the positive peak phase deviation in degrees.

Examples `READ:PM:PPOSITIVE?` might return `26.87`, indicating the positive peak phase deviation is 26.87 °.

READ:PM:PTPeak? (Query Only)

Returns the peak-peak phase deviation (Pk-Pk) in the PM measurement.

Conditions Measurement views: PM

Group Read commands

Syntax `READ:PM:PTPeak?`

Arguments None

Returns `<Pk-Pk> ::` `<NRf>` is the peak-peak phase deviation in degrees.

Examples `READ:PM:PTPEAK?` might return `46.34`, indicating the peak-peak phase deviation is 46.34 °.

READ:PM:RESult? (Query Only)

Returns the PM measurement results.

Conditions Measurement views: PM

Group Read commands

Syntax `READ:PM:RESult?`

Arguments None

Returns `<+Pk>,<–Pk>,<RMS>,<Pk-Pk>`

Where
`<+Pk> ::` `<NRf>` is the positive peak phase deviation in degrees.
`<-Pk> ::` `<NRf>` is the negative peak phase deviation in degrees.
`<RMS> ::` `<NRf>` is the RMS phase deviation in degrees.
`<Pk-Pk> ::` `<NRf>` is the peak-peak phase deviation in degrees.

Examples `READ:PM:RESULT?` might return `22.89,–23.45,15.12,46.34`.

READ:PM:RMS? (Query Only)

Returns the RMS phase deviation in the PM measurement.

Conditions Measurement views: PM

Group Read commands

Syntax `READ:PM:RMS?`

Arguments None

Returns `<RMS> ::` `<NRf>` is the RMS phase deviation in degrees.

Examples `READ:PM:RMS?` might return `15.12`, indicating the RMS frequency deviation is 15.12 °.

READ:PNOise:ALL? (Query Only)

Returns all results of the phase noise measurement.

Conditions Measurement views: Phase noise

Group Read commands

Syntax `READ:PNOise:ALL?`

Arguments None

Returns `<Cpower>,<Ferror>,<Pnoise>,<Tjitter>,<Rjitter>,<RFM>`

Where
`<Cpower> ::` `<NRf>` is the carrier power in dBm.
`<Ferror> ::` `<NRf>` is the frequency error in Hz.
`<Pnoise> ::` `<NRf>` is the RMS phase noise in degrees.
`<Tjitter> ::` `<NRf>` is the total jitter in seconds.
`<Rjitter> ::` `<NRf>` is the random jitter in seconds.
`<RFM> ::` `<NRf>` is the residual FM in Hz.

Examples `READ:PNOISE:ALL?` might return `-9.455,1.235E+6,51.43,2.312E-9,4.178E-9,14.58`, indicating
Carrier power: -9.455 dBm,
Frequency error: 1.235 MHz,
RMS phase noise: 51.43 °,
Total jitter: 2.312 ns,
Random jitter: 4.178 ns, and
Residual FM: 14.58 Hz.

READ:PNOise:CARRier:FERRor? (Query Only)

Returns the carrier frequency error in the phase noise measurement.

Conditions Measurement views: Phase noise

Group Read commands

Syntax `READ:PNOise:CARRier:FERRor?`

Arguments None

Returns `<NRf>` Carrier frequency error in Hz.

Examples `READ:PNOISE:CARRIER:FERROR?` might return `1.235E+6`, indicating that the carrier frequency error is 1.235 MHz.

READ:PNOise:CARRier:POWer? (Query Only)

Returns the carrier power in the phase noise measurement.

Conditions Measurement views: Phase noise

Group Read commands

Syntax `READ:PNOise:CARRier:POWer?`

Arguments None

Returns `<NRf>` Carrier power in dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `READ:PNOISE:CARRIER:POWER?` might return `-9.455`, indicating that the carrier power is -9.455 dBm.

READ:PNOise:JITTer? (Query Only)

Returns the jitter in the phase noise measurement.

Conditions Measurement views: Phase noise

Group Read commands

Syntax `READ:PNOise:JITTer?`

Arguments None

Returns `<NRf>` Jitter in seconds.

Examples `READ:PNOISE:JITTER?` might return `2.312E-9`, indicating that the jitter is 2.312 ns.

READ:PNOise:RESidual:FM? (Query Only)

Returns the residual FM in the phase noise measurement.

Conditions Measurement views: Phase noise

Group Read commands

Syntax `READ:PNOise:RESidual:FM?`

Arguments None

Returns `<NRf>` Residual FM in Hz.

Examples `READ:PNOISE:RESIDUAL:FM?` might return `14.58`, indicating that the residual FM is 14.58 Hz.

READ:PNOise:RMS:PNOise? (Query Only)

Returns the RMS phase noise in the phase noise measurement.

Conditions Measurement views: Phase noise

Group Read commands

Syntax `READ:PNOise:RMS:PNOise?`

Arguments None

Returns <NRf> RMS phase noise in degrees.

Examples READ:PNOISE:RMS:PNOISE? might return 51.43, indicating that the RMS phase noise is 51.43 °.

READ:PNOise:SPECtrum<x>:X? (Query Only)

Returns the frequencies of the specified trace.

The parameter <x> = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions Measurement views: Phase noise

Group Read commands

Syntax READ:PNOise:SPECtrum<x>:X?

Arguments None

Returns #<num_digit><num_byte><x(1)><x(2)>...<x(n)>

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<x(n)> is the frequency (Hz) at the n^{th} point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples READ:PNOISE:SPECTRUM1:X? might return #516020xxxx... (16020-byte data) for the frequencies of Trace 1.

READ:PNOise:SPECtrum<x>:XY? (Query Only)

Returns the frequency and phase noise pairs of the specified trace.

The parameter <x> = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions Measurement views: Phase noise

Group Read commands

Syntax `READ:PNOise:SPECtrum<x>:XY?`

Arguments None

Returns `#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>`

Where
<num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<x(n)><y(n)> is the frequency (Hz) and phase noise (dBc/Hz) pair at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:PNOISE:SPECTRUM1:XY?` might return `#516020xxxx...` (16020-byte data) for the frequency and phase noise pairs of the Trace 1.

READ:PNOise:SPECtrum<x>[:Y]? (Query Only)

Returns the phase noise values of the specified trace.

The parameter <x> = 1 and 2, representing Trace 1 and Trace 2, respectively.

Conditions Measurement views: Phase noise

Group Read commands

Syntax `READ:PNOise:SPECtrum<x>[:Y]?`

Arguments None

Returns `#<num_digit><num_byte><y(1)><y(2)>...<y(n)>`

Where <num_digit> is the number of digits in <num_byte>.
<num_byte> is the number of bytes of data that follow.
<y(n)> is the phase noise (dBc/Hz) at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:PNOISE:SPECTRUM1:Y` might return `#516020xxxx...` (16020-byte data) for the phase noise values of Trace 1.

READ:PULSe[:RESult]:ATX? (Query Only)

Returns the average transmitted power in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax `READ:PULSe[:RESult]:ATX?`

Arguments None

Returns `<first_pulse_num>,<ATX(1)>,< ATX(2)>,...<ATX(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<ATX(n)> ::` `<NRf>` is the average transmitted power for the pulse with the number of [first_pulse_num + n - 1] in dBm.
The unit can be changed to watts by the [SENSe]:POWer:UNITs command.
Volt is invalid in the average transmitted power measurement.

Examples `READ:PULSE:RESULT:ATX?` might return `0,-18.57,-18.73,-18.20,-18.53` for Pulse 0 to 3.

READ:PULSe[:RESult]:AVERage? (Query Only)

Returns the average on power in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax `READ:PULSe[:RESult]:AVERage?`

Arguments None

Returns `<first_pulse_num>,<avg(1)>,< avg(2)>,...<avg(n)>`

Where

<first_pulse_num> :: <NR1> is the first pulse number.
<avg(n)> :: <NRf> is the average on power for the pulse with the number of [first_pulse_num + n - 1] in dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples READ:PULSE:RESULT:AVERAGE? might return 0,-2.354,-2.368,-2.343,-2.358 for Pulse 0 to 3.

READ:PULSe[:RESult]:DRODb? (Query Only)

Returns the droop in dB in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax READ:PULSe[:RESult]:DRODb?

Related Commands READ:PULSe[:RESult]:DROop?

Arguments None

Returns <first_pulse_num>,<droop(1)>,<droop(2)>,...<droop(n)>

Where
<first_pulse_num> :: <NR1> is the first pulse number.
<droop(n)> :: <NRf> is the wattage droop for the pulse with the number of [first_pulse_num + n - 1] in dB.

Examples READ:PULSE:RESULT:DRODB? might return 1,-0.4,0.32,0.2,-0.1 for Pulse 1 to 4.

READ:PULSe[:RESult]:DROop? (Query Only)

Returns the average on power in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax `READ:PULSe[:RESult]:DROop?`

Related Commands READ:PULSe[:RESult]:DRODb?

Arguments None

Returns `<first_pulse_num>,<droop(1)>,<droop(2)>,...<droop(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<droop(n)> ::` `<NRf>` is the wattage droop for the pulse with the number of [first_pulse_num + n - 1] in percent (%).

Examples `READ:PULSE:RESULT:DROOP?` might return `0,-270.9E-3,-193.0E-3, -242.7E-3,-177.5E-3` for Pulse 0 to 3.

READ:PULSe[:RESult]:DUTPct? (Query Only)

Returns the duty factor (%) in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax `READ:PULSe[:RESult]:DUTPct?`

Arguments None

Returns `<first_pulse_num>,<duty_pct(1)>,<duty_pct(2)>,...`
`<duty_pct(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<duty_pct(n)> ::` `<NRf>` is the duty factor for the pulse with the number of [first_pulse_num + n - 1] in percent (%).

Examples READ:PULSE:RESULT:DUTPCT? might return 0,28.94,28.96,29.00,29.01 for Pulse 0 to 3.

READ:PULSe[:RESult]:DUTRatio? (Query Only)

Returns the duty factor (ratio) in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax READ:PULSe[:RESult]:DUTRatio?

Arguments None

Returns <first_pulse_num>,<duty_ratio(1)>,<duty_ratio(2)>,...<duty_ratio(n)>

Where
<first_pulse_num> :: <NR1> is the first pulse number.
<duty_ratio(n)> :: <NRf> is the duty factor for the pulse with the number of [first_pulse_num + n - 1] (no unit).

Examples READ:PULSE:RESULT:DUTRATIO? might return 0,289.4E-3,289.6E-3,290.0E-3,290.1E-3 for Pulse 0 to 3.

READ:PULSe[:RESult]:FALL? (Query Only)

Returns the fall time in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax READ:PULSe[:RESult]:FALL?

Arguments None

Returns <first_pulse_num>,<fall(1)>,<fall(2)>,...<fall(n)>

Where
<first_pulse_num> :: <NR1> is the first pulse number.
<fall(n)> :: <NRf> is the fall time for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples READ:PULSE:RESULT:FALL? might return 0,110.3E-9,90.45E-9, 95.03E-9,111.9E-9 for Pulse 0 to 3.

READ:PULSe[:RESult]:FDELta? (Query Only)

Returns the computed delta frequency from the carrier in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax READ:PULSe[:RESult]:FDELta?

Arguments None

Returns <first_pulse_num>,<freq_delta(1)>,<freq_delta(2)>,... <freq_delta(n)>

Where
<first_pulse_num> :: <NR1> is the first pulse number.
<freq_delta(n)> :: <NRf> is the frequency delta for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples READ:PULSE:RESULT:FDELTA? might return 1.0000000000,163.6692962646,82.8298492432,21.8995475769,89.296356201 for the first pulse and each pulse found.

READ:PULSe[:RESult]:FRDeviation? (Query Only)

Returns the frequency deviation in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax `READ:PULSe[:RESult]:FRDeviation?`

Arguments None

Returns `<first_pulse_num>,<freq_dev(1)>,<freq_dev(2)>,...<freq_dev(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<freq_dev(n)> ::` `<NRf>` is the frequency deviation for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples `READ:PULSE:RESULT:FRDEVIATION?` might return `1,740.6E+3,736.5E+3,718.3E+3,672.2E+3` for Pulse 1 to 4.

READ:PULSe[:RESult]:IRAMplitude? (Query Only)

Returns the pulse impulse response amplitude in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax `READ:PULSe[:RESult]:IRAMplitude?`

Returns `<first_pulse_num>,<amplitude(1)>,<amplitude(2)>,...<amplitude(n)>`

Where

`<first_pulse_num> ::` `<NR1>` is the first pulse number.

`<amplitude(n)> ::` `<NRf>` is the sidelobe level, in dB, from the mainlobe for the pulse with the number of [first_pulse_num + n - 1].

Examples `READ:PULSE[:RESULT]:IRAMPLITUDE?` might return `1.0000000000,-45.3,-47.5,-50.2` for Pulse 1 through 3.

READ:PULSe[:RESult]:IRTime? (Query Only)

Returns the impulse response time in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax `READ:PULSe[:RESult]:IRTime?`

Returns `<first_pulse_num>,<time(1)>,<time(2)>,...<time(n)>`

Where

`<first_pulse_num> ::` `<NR1>` is the first pulse number.

`<time(n)> ::` `<NRf>` is the sidelobe offset time from the mainlobe for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples `READ:PULSE[:RESULT]:IRTIME?` might return `1.0000000000,806.6708687693E-9,-846.6595318168E-9,-873.3477443457E-9` for Pulse 1 to 3.

READ:PULSe[:RESult]:MFReqerror? (Query Only)

Returns the maximum frequency error in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax `READ:PULSe[:RESult]:MFReqerror?`

Arguments None

Returns `<first_pulse_num>,<max_freq_err(1)>,<max_freq_err(2)>,...<max_freq_err(n)>`

Where

`<first_pulse_num> ::` `<NR1>` is the first pulse number.

<max_freq_err(n)> :: <NRf> is the maximum frequency error for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples READ:PULSE:RESULT:MFREQERROR? might return 1,597.5E+3,675.8E+3,642.8E+3,598.2E+3 for Pulse 1 to 4.

READ:PULSe[:RESult]:MPHerror? (Query Only)

Returns the maximum phase error in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax READ:PULSe[:RESult]:MPHerror?

Arguments None

Returns <first_pulse_num>,<max_phase_err(1)>,<max_phase_err(2)>,... <max_phase_err(n)>

Where
<first_pulse_num> :: <NR1> is the first pulse number.
<max_phase_err(n)> :: <NRf> is the maximum phase error for the pulse with the number of [first_pulse_num + n - 1] in degrees.

Examples READ:PULSE:RESULT:MPHERROR? might return 1,-9.221,-8.413, -11.853,-10.258 for Pulse 1 to 4.

READ:PULSe[:RESult]:OVEDb? (Query Only)

Returns the overshoot in dB in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax READ:PULSe[:RESult]:OVEDb?

Related Commands READ:PULSe[:RESult]:OVERshoot?

Arguments None

Returns `<first_pulse_num>,<overshoot(1)>,<overshoot(2)>,...<overshoot(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<overshoot(n)> ::` `<NRf>` is the overshoot for the pulse with the number of [first_pulse_num + n - 1] in dB.

Examples `READ:PULSE:RESULT:OVERDB?` might return `1,1.2,2.3,1.0,0.2` for Pulse 1 to 4.

READ:PULSe[:RESult]:OVERshoot? (Query Only)

Returns the overshoot in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax `READ:PULSe[:RESult]:OVERshoot?`

Related Commands READ:PULSe[:RESult]:OVEDb?

Arguments None

Returns `<first_pulse_num>,<overshoot(1)>,<overshoot(2)>,...<overshoot(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<overshoot(n)> ::` `<NRf>` is the overshoot for the pulse with the number of [first_pulse_num + n - 1] in percent (%) of watts or volts.

Examples `READ:PULSE:RESULT:OVERSHOOT?` might return `1,1.2,2.3,1.0,0.2` for Pulse 1 through 4.

READ:PULSe[:RESult]:PHDeviation? (Query Only)

Returns the phase deviation in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax `READ:PULSe[:RESult]:PHDeviation?`

Arguments None

Returns `<first_pulse_num>,<phase_dev(1)>,<phase_dev(2)>,...<phase_dev(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<phase_dev(n)> ::` `<NRf>` is the phase deviation for the pulse with the number of [first_pulse_num + n - 1] in degrees.

Examples `READ:PULSE:RESULT:PHDEVIATION?` might return `1,11.658,9.640,10.509,8.272` for Pulse 1 to 4.

READ:PULSe[:RESult]:PPFRequency? (Query Only)

Returns the pulse-pulse carrier frequency in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax `READ:PULSe[:RESult]:PPFRequency?`

Arguments None

Returns `<first_pulse_num>,<pp_freq(1)>,<pp_freq(2)>,...<pp_freq(n)>`

Where

<first_pulse_num> :: <NR1> is the first pulse number.
<pp_freq(n)> :: <NRf> is the pulse-pulse carrier frequency for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples READ:PULSE:RESULT:PPFREQUENCY? might return 0,0.000,1.258E+3,-3.121E+3,1.862E+3 for Pulse 0 to 3.

READ:PULSe[:RESult]:PPOWer? (Query Only)

Returns the peak power in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax READ:PULSe[:RESult]:PPOWer?

Arguments None

Returns <first_pulse_num>,<pk_power(1)>,<pk_power(2)>,...<pk_power(n)>

Where
<first_pulse_num> :: <NR1> is the first pulse number.
<pk_power(n)> :: <NRf> is the peak power for the pulse with the number of [first_pulse_num + n - 1] in dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples READ:PULSE:RESULT:PPOWER? might return 0,-2.26,-2.27,-2.23,-2.25 for Pulse 0 to 3.

READ:PULSe[:RESult]:PPPHase? (Query Only)

Returns the pulse-pulse carrier phase in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax `READ:PULSe[:RESult]:PPPHase?`

Arguments None

Returns `<first_pulse_num>,<pp_phase(1)>,<pp_phase(2)>,...<pp_phase(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<pp_phase(n)> ::` `<NRf>` is the pulse-pulse carrier phase for the pulse with the number of [first_pulse_num + n - 1] in degrees.

Examples `READ:PULSE:RESULT:PPPHASE?` might return `0,0.000,21.66,46.76,57.56` for Pulse 0 to 3.

READ:PULSe[:RESult]:RINTerval? (Query Only)

Returns the repetition interval in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax `READ:PULSe[:RESult]:RINTerval?`

Arguments None

Returns `<first_pulse_num>,<rep_int(1)>,<rep_int(2)>,...<rep_int(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<rep_int(n)> ::` `<NRf>` is the repetition interval for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples `READ:PULSE:RESULT:RINTERVAL?` might return `0,16.03E-6,16.08E-6, 16.07E-6,16.02E-6` for Pulse 0 to 3.

READ:PULSe[:RESult]:RIPDb? (Query Only)

Returns the ripple in dB in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax `READ:PULSe[:RESult]:RIPDb?`

Related Commands READ:PULSe[:RESult]:RIPPle?

Arguments None

Returns `<first_pulse_num>,<ripple(1)>,<ripple(2)>,...<ripple(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<ripple(n)> ::` `<NRf>` is the ripple for the pulse with the number of [first_pulse_num + n - 1] in dB.

Examples `READ:PULSE:RESULT:RIPDB?` might return `1,0.4,0.32,0.2,0.1` for Pulse 1 to 4.

READ:PULSe[:RESult]:RIPPle? (Query Only)

Returns the ripple in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax `READ:PULSe[:RESult]:RIPPle?`

Related Commands READ:PULSe[:RESult]:RIPDb?

Arguments None

Returns `<first_pulse_num>,<ripple(1)>,<ripple(2)>,...<ripple(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.

<ripple(n)> :: <NRf> is the watt or voltage ripple for the pulse with the number of [first_pulse_num + n - 1] in percent (%).

Examples READ:PULSE:RESULT:RIPPLE? might return 0,106.5E-3,177.6E-3, 148.3E-3,148.5E-3 for Pulse 0 to 3.

READ:PULSe[:RESult]:RISE? (Query Only)

Returns the rise time in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax READ:PULSe[:RESult]:RISE?

Arguments None

Returns <first_pulse_num>,<rise(1)>,<rise(2)>,...<rise(n)>

Where
<first_pulse_num> :: <NR1> is the first pulse number.
<rise(n)> :: <NRf> is the rise time for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples READ:PULSE:RESULT:RISE? might return 0,92.94E-9,115.9E-9, 115.1E-9,97.45E-9 for Pulse 0 to 3.

READ:PULSe[:RESult]:RMSFreqerror? (Query Only)

Returns the RMS frequency error in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax READ:PULSe[:RESult]:RMSFreqerror?

Arguments None

Returns <first_pulse_num>,<RMS_freq_err(1)>,<RMS_freq_err(2)>,...<RMS_freq_err(n)>

Where
<first_pulse_num> :: <NR1> is the first pulse number.
<RMS_freq_err(n)> :: <NRf> is the RMS frequency error for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples READ:PULSE:RESULT:RMSFREQERROR? might return 1,51.54E+3,69.20E+3,64.21E+3,51.02E+3 for Pulse 1 to 4.

READ:PULSe[:RESult]:RMSPherror? (Query Only)

Returns the RMS phase error in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax READ:PULSe[:RESult]:RMSPherror?

Arguments None

Returns <first_pulse_num>,<RMS_phase_err(1)>,<RMS_phase_err(2)>,...<RMS_phase_err(n)>

Where
<first_pulse_num> :: <NR1> is the first pulse number.
<RMS_phase_err(n)> :: <NRf> is the RMS phase error for the pulse with the number of [first_pulse_num + n - 1] in degrees.

Examples READ:PULSE:RESULT:RMSPHERROR? might return 1,908.4E-3,752.8E-3,981.7E-3,886.4E-3 for Pulse 1 to 4.

READ:PULSe[:RESult]:RRATe? (Query Only)

Returns the repetition rate in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax `READ:PULSe[:RESult]:RRATe?`

Arguments None

Returns `<first_pulse_num>,<rep_rate(1)>,<rep_rate(2)>,...<rep_rate(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<rep_rate(n)> ::` `<NRf>` is the repetition rate for the pulse with the number of [first_pulse_num + n - 1] in Hz.

Examples `READ:PULSE:RESULT:RRATE?` might return `0,62.50E+3,62.52E+3,62.51E+3,62.49E+3` for Pulse 0 to 3.

READ:PULSe[:RESult]:TIME? (Query Only)

Returns the time in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax `READ:PULSe[:RESult]:TIME?`

Arguments None

Returns `<first_pulse_num>,<time(1)>,<time(2)>,...<time(n)>`

Where
`<first_pulse_num> ::` `<NR1>` is the first pulse number.
`<time(n)> ::` `<NRf>` is the time for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples READ:PULSE:RESULT:TIME? might return 1,7.937E-3,8.436E-3, 6.504E-3,9.876E-3 for Pulse 1 to 4.

READ:PULSe[:RESult]:WIDTh? (Query Only)

Returns the pulse width in the pulse table.

Conditions Measurement views: Pulse table

Group Read commands

Syntax READ:PULSe[:RESult]:WIDTh?

Arguments None

Returns <first_pulse_num>,<width(1)>,<width(2)>,...<width(n)>

Where <first_pulse_num> :: <NR1> is the first pulse number.
<width(n)> :: <NRf> is the pulse width for the pulse with the number of [first_pulse_num + n - 1] in seconds.

Examples READ:PULSE:RESULT:WIDTH? might return 0,4.630E-6,4.632E-6, 4.639E-6,4.642E-6 for Pulse 0 to 3.

READ:PULSe:STATistics? (Query Only)

Returns the trace data of the pulse statistics measurement selected by the DISPlay:PULSe:SELect:RESult command.

NOTE. *Select the plot type (Trend, FFT, Time Trend or Histogram) using the DISPlay:PULSe:STATistics:PLOT command before executing this query.*

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax READ:PULSe:STATistics?

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the amplitude at the n^{th} data point.
The unit is current measurement unit (Plot = Trend or Time Trend), dB (Plot = FFT), count or percent (Plot = Histogram)
4-byte little endian floating-point format specified in IEEE 488.2.

The unit of power is selected by the [SENSe]:POWer:UNITs command.

Examples `READ:PULSE:STATISTICS?` might return `#264xxxx...` (64-byte data) for the statistics trace of the pulse width measurement when DISPlay:PULSe:SELect:RESult is set to WIDTh.

READ:PULSe:STATistics:ATX? (Query Only)

Returns the average transmitted power in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:ATX?`

Arguments None

Returns `<ATX_avg>,<ATX_min>,<ATX_max>`

Where
`<ATX_avg>` :: `<NRf>` is the average of the average transmitted power.
`<ATX_min>` :: `<NRf>` is the minimum of the average transmitted power.
`<ATX_max>` :: `<NRf>` is the maximum of the average transmitted power.
Unit: dBm.

The unit can be changed to watts by the [SENSe]:POWer:UNITs command. Volt is invalid in the average transmitted power measurement.

Examples `READ:PULSE:STATISTICS:ATX?` might return `-18.51,-18.74,-18.12` for the average transmitted power in the pulse statistics.

READ:PULSe:STATistics:AVERage? (Query Only)

Returns the average on power in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:AVERage?`

Arguments None

Returns `<avg_avg>,<avg_min>,< avg_max>`

Where
`<avg_avg> ::` `<NRf>` is the average of the average on power.
`<avg_min> ::` `<NRf>` is the minimum of the average on power.
`<avg_max> ::` `<NRf>` is the maximum of the average on power.
Unit: dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `READ:PULSE:STATISTICS:AVERAGE?` might return `-2.35,-2.36,-2.34` for the average on power in the pulse statistics.

READ:PULSe:STATistics:DRODb? (Query Only)

Returns the droop in dB in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:DRODb?`

Related Commands READ:PULSe:STATistics:DROop?

Arguments None

Returns `<droop_avg>,<droop_min>,<droop_max>`

Where
`<droop_avg>` :: `<NRf>` is the average droop.
`<droop_min>` :: `<NRf>` is the minimum droop.
`<droop_max>` :: `<NRf>` is the maximum droop.
Unit: in dB by watts.

Examples `READ:PULSE:STATISTICS:DRODB?` might return `22.67E-3,-613.5E-3, 633.8E-3` for the droop in the pulse statistics.

READ:PULSe:STATistics:DROop? (Query Only)

Returns the droop in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:DROop?`

Related Commands READ:PULSe:STATistics:DRODb?

Arguments None

Returns `<droop_avg>,<droop_min>,<droop_max>`

Where
`<droop_avg>` :: `<NRf>` is the average droop.
`<droop_min>` :: `<NRf>` is the minimum droop.
`<droop_max>` :: `<NRf>` is the maximum droop.
Unit: Percent (%) by watts.

Examples `READ:PULSE:STATISTICS:DROOP?` might return `22.67E-3,-613.5E-3, 633.8E-3` for the droop in the pulse statistics.

READ:PULSe:STATistics:DUTPct? (Query Only)

Returns the duty factor (%) in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:DUTPct?`

Related Commands

Arguments None

Returns `<duty_pct_avg>,<duty_pct_min)>,<duty_pct_max>`

Where
`<duty_pct_avg>` :: `<NRf>` is the average duty factor.
`<duty_pct_min>` :: `<NRf>` is the minimum duty factor.
`<duty_pct_max>` :: `<NRf>` is the maximum duty factor.
Unit: Percent (%).

Examples `READ:PULSE:STATISTICS:DUTPCT?` might return `2.437,2.310,2.657` for the duty factor (%) in the pulse statistics.

READ:PULSe:STATistics:DUTRatio? (Query Only)

Returns the duty factor (ratio) in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:DUTRatio?`

Arguments None

Returns <duty_ratio_avg>,<duty_ratio_min>,<duty_ratio_max)>

Where
<duty_ratio_avg> :: <NRf> is the average duty factor.
<duty_ratio_min> :: <NRf> is the minimum duty factor.
<duty_ratio_max> :: <NRf> is the maximum duty factor.
Unit: None.

Examples READ:PULSE:STATISTICS:DUTRATIO? might return 24.37E-3,23.11E-3, 26.57E-3 for the duty factor (ratio) in the pulse statistics.

READ:PULSe:STATistics:FALL? (Query Only)

Returns the fall time in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax READ:PULSe:STATistics:FALL?

Arguments None

Returns <fall_avg>,<fall_min>,<fall_max>

Where
<fall_avg> :: <NRf> is the average fall time.
<fall_min> :: <NRf> is the minimum fall time.
<fall_max> :: <NRf> is the maximum fall time.
Unit: Seconds.

Examples READ:PULSE:STATISTICS:FALL? might return 70.27E-9,69.62E-9, 71.27E-9 for the fall time in the pulse statistics.

READ:PULSe:STATistics:FDELta? (Query Only)

Return the delta frequency measurement in the pulse statistics.

Group Read commands

Syntax `READ:PULSe:STATistics:FDELta?`

Arguments None

Returns `<freq_delta_avg>,<freq_delta_min>,<freq_delta_max>`

Where
`<freq_delta_avg> ::` `<NRf>` is the average frequency deviation.
`<freq_delta_min> ::` `<NRf>` is the minimum frequency deviation.
`<freq_delta_max> ::` `<NRf>` is the maximum frequency deviation.
Unit: Hz.

Examples `READ:PULSE:STATISTICS:FDELTA?` might return `77.9785690308,-110.1940994263,287.8957519531` for the frequency delta in the pulse statistics.

READ:PULSe:STATistics:FRDeviation? (Query Only)

Returns the frequency deviation in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:FRDeviation?`

Arguments None

Returns `<freq_dev_avg>,<freq_dev_min>,<freq_dev_max>`

Where
`<freq_dev_avg> ::` `<NRf>` is the average frequency deviation.
`<freq_dev_min> ::` `<NRf>` is the minimum frequency deviation.
`<freq_dev_max> ::` `<NRf>` is the maximum frequency deviation.
Unit: Hz.

Examples `READ:PULSE:STATISTICS:FRDEVIATION?` might return `754.1E+3,660.5E+3,835.2E+3` for the frequency deviation in the pulse statistics.

READ:PULSe:STATistics:IRAMplitude? (Query Only)

Returns the Pulse Impulse Response amplitude measurement in the pulse statistics.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:IRAMplitude?`

Returns `<amp_avg>,<amp_min>,<amp_max>`

Where

`<amp_avg> ::` `<NRf>` is the average of the impulse response amplitude.

`<amp_min> ::` `<NRf>` is the minimum of the impulse response amplitude.

`<amp_max> ::` `<NRf>` is the maximum of impulse response amplitude.

Unit: dB.

Examples `READ:PULSE:STATISTICS:IRAMPLITUDE?` might return `-45.3,-47.1,-43.2` for the impulse response amplitude in the pulse statistics.

READ:PULSe:STATistics:IRTime? (Query Only)

Returns the impulse response time in the pulse statistics.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:IRTime?`

Returns `<avg_avg>,<avg_min>,<avg_max>`

Where

<avg_avg> :: <NRf> is the average of the impulse response time.

<avg_min> :: <NRf> is the minimum of the impulse response time.

<avg_max> :: <NRf> is the maximum of impulse response time.

Unit: seconds.

Examples READ:PULSE:STATISTICS:IRTIME? might return -2.35,-2.36,-2.34 for the impulse response time in the pulse statistics.

READ:PULSe:STATistics:MFReqerror? (Query Only)

Returns the maximum frequency error in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax READ:PULSe:STATistics:MFReqerror?

Arguments None

Returns <max_freq_err_avg>,<max_freq_err_min>,<max_freq_err_max>

Where

<max_freq_err_avg> :: <NRf> is the average of maximum frequency error.

<max_freq_err_min> :: <NRf> is the minimum of maximum frequency error.

<max_freq_err_max> :: <NRf> is the maximum of maximum frequency error.

Unit: Hz.

Examples READ:PULSE:STATISTICS:MFREQERROR? might return 645.0E+3,555.6E+3,738.8E+3 for the maximum frequency error in the pulse statistics.

READ:PULSe:STATistics:MPHerror? (Query Only)

Returns the maximum phase error in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:MPHerror?`

Arguments None

Returns `<max_phase_err_avg>,<max_phase_err_min>,<max_phase_err_max>`

Where
`<max_phase_err_avg>` :: `<NRf>` is the average of maximum phase error.
`<max_phase_err_min>` :: `<NRf>` is the minimum of maximum phase error.
`<max_phase_err_max>` :: `<NRf>` is the maximum of maximum phase error.
Unit: Degrees.

Examples `READ:PULSE:STATISTICS:MPHERROR?` might return `-11.47,-17.18,-7.61` for the maximum phase error in the pulse statistics.

READ:PULSe:STATistics:OVEDb? (Query Only)

Returns the overshoot measurement result in dB in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd or TTRend.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:OVEDb?`

Related Commands READ:PULSe:STATistics:OVERshoot?

Arguments None

Returns `<overshoot_avg>,<overshoot_min>,<overshoot_max>`

Where
`<overshoot_avg>` :: `<NRf>` is the average overshoot.
`<overshoot_min>` :: `<NRf>` is the minimum overshoot.
`<overshoot_max>` :: `<NRf>` is the maximum overshoot.
Unit: dB.

Examples `READ:PULSE:STATISTICS:OVEDB?` might return `0.3,0.1,0.5` for the overshoot measurement result in dB in the pulse statistics.

READ:PULSe:STATistics:OVERshoot? (Query Only)

Returns the overshoot in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd or TTRend.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:OVERshoot?`

Related Commands READ:PULSe:STATistics:OVEDb?

Arguments None

Returns `<overshoot_avg>,<overshoot_min>,<overshoot_max>`

Where
`<overshoot_avg>` :: `<NRf>` is the average overshoot.
`<overshoot_min>` :: `<NRf>` is the minimum overshoot.
`<overshoot_max>` :: `<NRf>` is the maximum overshoot.
Unit: Percent (%) as watts or volts.

Examples `READ:PULSE:STATISTICS:OVERSHOOT?` might return `0.3,0.1,0.5` for the overshoot in the pulse statistics.

READ:PULSe:STATistics:PHDeviation? (Query Only)

Returns the phase deviation in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:PHDeviation?`

Arguments None

Returns `<phase_dev_avg>,<phase_dev_min>,<phase_dev_max>`

Where
`<phase_dev_avg>` :: `<NRf>` is the average phase deviation.
`<phase_dev_min>` :: `<NRf>` is the minimum phase deviation.
`<phase_dev_max>` :: `<NRf>` is the maximum phase deviation.
Unit: Degrees.

Examples `READ:PULSE:STATISTICS:PHDEVIATION?` might return `11.678,7.694,17.374` for the phase deviation in the pulse statistics.

READ:PULSe:STATistics:PPFRequency? (Query Only)

Returns the pulse-pulse carrier frequency in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:PPFRequency?`

Arguments None

Returns `<pp_freq_avg>,<pp_freq_min)>,<pp_freq_max>`

Where
<pp_freq_avg> :: <NRf> is the average pulse-pulse carrier frequency.
<pp_freq_min> :: <NRf> is the minimum pulse-pulse carrier frequency.
<pp_freq_max> :: <NRf> is the maximum pulse-pulse carrier frequency.
Unit: Hz.

Examples READ:PULSE:STATISTICS:PPFREQUENCY? might return 1.135E+3,311.3E+3,-262.8E+3 for the pulse-pulse carrier frequency in the pulse statistics.

READ:PULSe:STATistics:PPOWer? (Query Only)

Returns the peak power in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax READ:PULSe:STATistics:PPOWer?

Arguments None

Returns <pk_power_avg>,<pk_power_min>,<pk_power_max>

Where
<pk_power_avg> :: <NRf> is the average peak power.
<pk_power_min> :: <NRf> is the minimum peak power.
<pk_power_max> :: <NRf> is the maximum peak power.
Unit: dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples READ:PULSe:STATistics:PPOWer? might return -2.273,-2.313,-2.235 for the peak power in the pulse statistics.

READ:PULSe:STATistics:PPPHase? (Query Only)

Returns the pulse-pulse carrier phase in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:PPPHase?`

Arguments None

Returns `<pp_phase_avg>,<pp_phase_min>,<pp_phase_max>`

Where
`<pp_phase_avg> ::` `<NRf>` is the average pulse-pulse carrier phase.
`<pp_phase_min> ::` `<NRf>` is the minimum pulse-pulse carrier phase.
`<pp_phase_max> ::` `<NRf>` is the maximum pulse-pulse carrier phase.
Unit: Degrees.

Examples `READ:PULSE:STATISTICS:PPPHASE?` might return `-9.298E-3,-254.3E-3,311.7E-3` for the pulse-pulse carrier phase in the pulse statistics.

READ:PULSe:STATistics:RINTerval? (Query Only)

Returns the repetition interval in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:RINTerval?`

Arguments None

Returns `<rep_int_avg>,<rep_int_min>,<rep_int_max>`

Where
`<rep_int_avg> ::` `<NRf>` is the average repetition interval.
`<rep_int_min> ::` `<NRf>` is the minimum repetition interval.
`<rep_int_max> ::` `<NRf>` is the maximum repetition interval.

Unit: Seconds.

Examples `READ:PULSE:STATISTICS:RINTERVAL?` might return `240.5E-6,217.9E-6,281.2E-6` for the repetition interval in the pulse statistics.

READ:PULSe:STATistics:RIPDb? (Query Only)

Returns the ripple in dB in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:RIPDb?`

Related Commands READ:PULSe:STATistics:RIPPle?

Arguments None

Returns `<ripple_avg>,<ripple_min>,<ripple_max>`

Where
`<ripple_avg> ::` `<NRf>` is the average ripple.
`<ripple_min> ::` `<NRf>` is the minimum ripple.
`<ripple_max> ::` `<NRf>` is the maximum ripple.
Unit: dB.

Examples `READ:PULSE:STATISTICS:RIPDB?` might return `160.4E-3,83.78E-3,287.7E-3` for the ripple in dB in the pulse statistics.

READ:PULSe:STATistics:RIPPle? (Query Only)

Returns the ripple in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:RIPPle?`

Related Commands READ:PULSe:STATistics:RIPDb?

Arguments None

Returns `<ripple_avg>,<ripple_min>,<ripple_max>`

Where
`<ripple_avg> ::` `<NRf>` is the average ripple.
`<ripple_min> ::` `<NRf>` is the minimum ripple.
`<ripple_max> ::` `<NRf>` is the maximum ripple.
Unit: Percent (%) as watts or volts.

Examples `READ:PULSE:STATISTICS:RIPPLE?` might return `160.4E-3,83.78E-3,287.7E-3` for the ripple in the pulse statistics.

READ:PULSe:STATistics:RISE? (Query Only)

Returns the rise time in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:RISE?`

Arguments None

Returns `<rise_avg>,<rise_min>,<rise_max>`

Where
`<rise_avg> ::` `<NRf>` is the average rise time.
`<rise_min> ::` `<NRf>` is the minimum rise time.
`<rise_max> ::` `<NRf>` is the maximum rise time.
Unit: Seconds.

Examples `READ:PULSE:STATISTICS:RISE?` might return `105.4E-9,91.65E-9, 116.2E-9` for the rise time in the pulse statistics.

READ:PULSe:STATistics:RMSFreqerror? (Query Only)

Returns the RMS frequency error in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:RMSFreqerror?`

Arguments None

Returns `<RMS_freq_err_avg>,<RMS_freq_err_min>,<RMS_freq_err_max>`

Where
`<RMS_freq_err_avg>` :: `<NRf>` is the average of RMS frequency error.
`<RMS_freq_err_min>` :: `<NRf>` is the minimum of RMS frequency error.
`<RMS_freq_err_max>` :: `<NRf>` is the maximum of RMS frequency error.
Unit: Hz.

Examples `READ:PULSE:STATISTICS:RMSFREQERROR?` might return `63.67E+3,45.49E+3,81.28E+3` for the RMS frequency error in the pulse statistics.

READ:PULSe:STATistics:RMSPherror? (Query Only)

Returns the RMS phase error in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:RMSPherror?`

Arguments None

Returns <RMS_phase_err_avg>,<RMS_phase_err_min>,<RMS_phase_err_max>

Where
<RMS_phase_err_avg> :: <NRf> is the average of RMS phase error.
<RMS_phase_err_min> :: <NRf> is the minimum of RMS phase error.
<RMS_phase_err_max> :: <NRf> is the maximum of RMS phase error.
Unit: Degrees.

Examples READ:PULSE:STATISTICS:RMSPHERROR? might return 1.032,604.5E-3, 1.606 for the RMS phase error in the pulse statistics.

READ:PULSe:STATistics:RRATe? (Query Only)

Returns the repetition rate in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax READ:PULSe:STATistics:RRATe?

Arguments None

Returns <rep_rate_avg>,<rep_rate_min>,<rep_rate_max>

Where
<rep_rate_avg> :: <NRf> is the average repetition rate.
<rep_rate_min> :: <NRf> is the minimum repetition rate.
<rep_rate_max> :: <NRf> is the maximum repetition rate.
Unit: Hz.

Examples READ:PULSE:STATISTICS:RRATE? might return 62.50E+3,62.49E+3,62.52E+3 for the repetition rate in the pulse statistics.

READ:PULSe:STATistics:WIDTh? (Query Only)

Returns the pulse width in the pulse statistics. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to TRENd.

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:WIDTh?`

Arguments None

Returns `<width_avg>,<width_min>,<width_max>`

Where
`<width_avg>` :: `<NRf>` is the average pulse width.
`<width_min>` :: `<NRf>` is the minimum pulse width.
`<width_max>` :: `<NRf>` is the maximum pulse width.
Unit: Seconds.

Examples `READ:PULSE:STATISTICS:WIDTH?` might return `4.636E-6,4.630E-6,4.643E-6` for the pulse width in the pulse statistics.

READ:PULSe:STATistics:X? (Query Only)

Returns the horizontal values of the trace data of the pulse statistics measurement selected by the DISPlay:PULSe:SELect:RESult command.

NOTE. *Select the plot type (Trend, FFT, Time Trend or Histogram) using the DISPlay:PULSe:STATistics:PLOT command before executing this query.*

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:X?`

Arguments None

Returns `#<num_digit><num_byte><data1><data2>....data<n>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the horizontal value of the statistics graph at the n^{th} data point.

The unit is trace number (Plot = Trend), Hz (Plot = FFT), second (Plot = Time Trend) or current measurement unit (Plot = Histogram).
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:PULSE:STATISTICS:X?` might return `#264xxxx...` (64-byte data) for the statistics trace of the pulse width measurement when DISPlay:PULSe:SELect:RESult is set to WIDTh.

READ:PULSe:STATistics:XY? (Query Only)

Returns the horizontal and vertical values of the trace data of the pulse statistics measurement selected by the DISPlay:PULSe:SELect:RESult command.

NOTE. *Select the plot type (Trend, FFT, Time Trend or Histogram) using the DISPlay:PULSe:STATistics:PLOT command before executing this query.*

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics:XY?`

Arguments None

Returns `#<num_digit><num_byte><data1><data2>....data<n>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the horizontal value of the statistics graph at the n^{th} data point.
`<data(n+1)>` is the vertical value of the statistics graph at the n^{th} data point.

For horizontal values, the unit is trace number (Plot = Trend), Hz (Plot = FFT), second (Plot = Time Trend) or current measurement unit (Plot = Histogram).

For vertical values, the unit is current measurement unit (Plot = Trend or Time Trend), dB (Plot = FFT), count (Plot = Histogram)
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:PULSE:STATISTICS:XY?` might return `#264xxxx...` (64-byte data) for the statistics trace of the pulse width measurement when DISPlay:PULSe:SELect:RESult is set to WIDTh.

READ:PULSe:STATistics[:Y]? (Query Only)

Returns the trace data of the pulse statistics measurement selected by the DISPlay:PULSe:SELect:RESult command.

NOTE. *Select the plot type (Trend, FFT, Time Trend or Histogram) using the DISPlay:PULSe:STATistics:PLOT command before executing this query.*

Conditions Measurement views: Pulse statistics

Group Read commands

Syntax `READ:PULSe:STATistics[:Y]?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the amplitude at the n^{th} data point.
The unit is current measurement unit (Plot = Trend or Time Trend), dB (Plot = FFT), count or percent (Plot = Histogram)
4-byte little endian floating-point format specified in IEEE 488.2.

The unit of power is selected by the [SENSe]:POWer:UNITs command.

READ:PULSe:TRACe:X? (Query Only)

Returns the time values of the pulse trace. Use the DISPlay:PULSe:SELect:NUMBer command to select the pulse, and the DISPlay:PULSe:SELect:RESult command to select the measurement result.

Conditions Measurement views: Pulse trace

Group Read commands

Syntax `READ:PULSe:TRACe:X?`

Arguments None

Returns `#<num_digit><num_byte><X(1)><X(2)>...<X(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<X(n)>` is the time in seconds at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:PULSE:TRACE:X?` might return `#43204xxxx...` (3204-byte data) for the time values of the trace.

READ:PULSe:TRACe:XY? (Query Only)

Returns the horizontal (time) and vertical value pairs of the pulse trace. Use the DISPlay:PULSe:SELect:NUMBer command to select the pulse, and the DISPlay:PULSe:SELect:RESult command to select the measurement result.

Conditions Measurement views: Pulse trace

Group Read commands

Syntax `READ:PULSe:TRACe:XY?`

Arguments None

Returns `#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<x(n)><y(n)>` is the horizontal value (time in seconds) and vertical value pair at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

The vertical unit depends on measurement results: Hz for frequency error and deviation, degrees for phase error and deviation, otherwise dBm. The vertical unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `READ:PULSE:TRACE:XY?` might return `#43204xxxx...` (3204-byte data) for the horizontal (time) and vertical value pairs of the pulse trace.

READ:PULSe:TRACe[:Y]? (Query Only)

Returns the vertical values of the pulse trace. Use the DISPlay:PULSe:SELect:NUMBer command to select the pulse, and the DISPlay:PULSe:SELect:RESult command to select the measurement result.

Conditions Measurement views: Pulse trace

Group Read commands

Syntax `READ:PULSe:TRACe[:Y]?`

Arguments None

Returns `#<num_digit><num_byte><y(1)><y(2)>...<y(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<y(n)>` is the vertical value of the pulse trace at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

The unit depends on measurement results: Hz for frequency error and deviation, degrees for phase error and deviation, otherwise dBm. The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples READ:PULSE:TRACE:Y? might return #43204xxxx... (3204-byte data) for the vertical values of the pulse trace.

READ:SEM:MHITs:COUNt? (Query Only)

Returns the number of rows in the results table in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Read commands

Syntax READ:SEM:MHITs:COUNt?

Arguments None

Returns <NRf> The number of rows in the table.

Examples READ:SEM:MHITS:COUNT? might return 7 indicating there are seven rows in the results table.

READ:SEM:MHITs<x>:AMPLitude:ABSolute? (Query Only)

Returns the absolute amplitude of the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Read commands

Syntax READ:SEM:MHITs<x>:AMPLitude:ABSolute?

Arguments None

Returns <NRf> Absolute amplitude of the specified signal in dBm.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples `READ:SEM:MHITs2:AMPLitude:ABSolute?` might return `–73.17` indicating that the absolute amplitude of signal in row 2 is –73.17 dBm.

READ:SEM:MHITs<x>:AMPLitude:RELative? (Query Only)

Returns the relative amplitude of the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Read commands

Syntax `READ:SEM:MHITs<x>:AMPLitude:RELative?`

Arguments None

Returns <NRf> Relative amplitude of the specified signal in dBm.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples `READ:SEM:MHITs2:AMPLitude:RELative?` might return `–91.81` indicating that the relative amplitude of mask hit signal in row 2 is –91.81 dBm.

READ:SEM:MHITs<x>:FREQuency? (Query Only)

Returns the frequency of the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Read commands

Syntax `READ:SEM:MHITs<x>:FREQuency?`

Arguments None

Returns <NRf> Frequency of the specified signal in Hz.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples `READ:SEM:MHITs4:FREQuency?` might return `2.235E+6` indicating the frequency for mask hit signal in row 4 is 2.235 MHz.

READ:SEM:MHITs<x>:INTeg:ABSolute? (Query Only)

Returns the absolute integration amplitude of the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Read commands

Syntax `READ:SEM:MHITs<x>:INTeg:ABSolute?`

Arguments None

Returns <NRf> Absolute integration amplitude of the specified signal in dBm.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples `READ:SEM:MHITs4:INTeg:ABSolute?` might return `–75.14` indicating that the absolute integration amplitude for mask hit signal in row 4 is –75.14 dBM.

READ:SEM:MHITs<x>:INTeg:RELative? (Query Only)

Returns the relative integration amplitude of the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Read commands

Syntax `READ:SEM:MHITs<x>:INTeg:RELative?`

Arguments None

Returns <NRf> Relative integration amplitude of the specified signal in dBm.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples READ:SEM:MHITs4:INTeg:RELative? might return –85.14 indicating that the relative integration amplitude for mask hit signal in row 4 is –85.14 dBM.

READ:SEM:MHITs<x>:MARGin:ABSolute? (Query Only)

Returns the absolute margin for the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Read commands

Syntax READ:SEM:MHITs<x>:MARGin:ABSolute?

Arguments None

Returns <NRf> Absolute margin of the specified signal in dB.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples READ:SEM:MHITs4:MARGin:ABSolute? might return –62.33 indicating that the absolute margin for mask hit signal in row 4 is –62.33 dB.

READ:SEM:MHITs<x>:MARGin:RELative? (Query Only)

Returns the relative margin for the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Read commands

Syntax READ:SEM:MHITs<x>:MARGin:RELative?

Arguments None

Returns <NRf> Relative margin of the specified signal in dB.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples READ:SEM:MHITs4:MARGin:RELative? might return –62.33 indicating that the relative margin for mask hit signal in row 4 is –62.33 dB.

READ:SEM:MHITs<x>:OFFSet? (Query Only)

Returns the offset for the specified mask hit signal in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Read commands

Syntax READ:SEM:MHITs<x>:OFFSet?

Arguments None

Returns <NRf> specifies the mask hit offset.

<x> specifies the row in the results table, where the range is from 1 to 12.

Examples READ:SEM:MHITs4:OFFSet? might return BL indicating the offset for mask hit signal in row 4 is BL.

READ:SEM:PASS? (Query Only)

Returns the pass/fail limit test result in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Read commands

Syntax READ:SEM:PASS?

Arguments None

Returns 0 (fail) or 1 (pass)

Examples `READ:SEM:PASS?` might return 1, indicating that the limit test was successful.

READ:SEM:REF:POWer? (Query Only)

Returns the reference power level in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Read commands

Syntax `READ:SEM:REF:POWer?`

Arguments None

Returns `<NRf>` specifies the reference power.

Examples `READ:SEM:REF:POWER?` might return `–16.92 dBm` indicating that the reference power level is –16.92 dBm.

READ:SEM:SPECtrum:X? (Query Only)

Returns the frequencies of the spectrum trace in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Read commands

Syntax `READ:SEM:SPECtrum:X?`

Arguments None

Returns `#<num_digit><num_byte><x(1)><x(2)>...<x(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<x(n)>` is the frequency (Hz) at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:SEM:SPECTRUM:X?` might return `#516020xxxx...` (16020-byte data) for the frequencies of the spectrum trace.

READ:SEM:SPECtrum:XY? (Query Only)

Returns the frequency and amplitude pairs of the spectrum trace in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Read commands

Syntax `READ:SEM:SPECtrum:XY?`

Arguments None

Returns `#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<x(n)><y(n)>` is the frequency (Hz) and amplitude (dBm) pair at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:SEM:SPECTRUM:XY?` might return `#516020xxxx...` (16020-byte data) for the frequency and amplitude pairs of the spectrum trace.

READ:SEM:SPECtrum[:Y]? (Query Only)

Returns the amplitude of the spectrum trace in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Read commands

Syntax `READ:SEM:SPECtrum[:Y]?`

Arguments None

Returns `#<num_digit><num_byte><x(1)><x(2)>...<x(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<x(n)>` is the frequency (Hz) at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:SEM:SPECTRUM[:Y]?` might return `#516020xxxx...` (16020-byte data) for the frequencies of the spectrum trace.

READ:SGRam? (Query Only)

Returns the spectrogram trace data.The line is selected using the TRACe:SGRam:SELect:LINE command.

Conditions Measurement views: Spectrogram

Group Read commands

Syntax `READ:SGRam?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the trace data in dBm for the point n,
4-byte little endian floating-point format specified in IEEE 488.2.

The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `READ:SGRAM?` might return `#43204xxxx...` (3204-byte data) for the spectrogram trace.

READ:SPECtrum:TRACe<x>? (Query Only)

Returns the trace data in the Spectrum measurement.

The parameter <x> = 1 to 5.

NOTE. *TRACe5 (spectrogram) is valid when the spectrum and spectrogram measurements are running.*

Conditions Measurement views: Spectrum

Group Read commands

Syntax `READ:SPECtrum:TRACe<x>?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the trace data in dBm for the point n,
4-byte little endian floating-point format specified in IEEE 488.2.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `READ:SPECTRUM:TRACE1` might return `#43204xxxx...` (3204-byte data) for Trace 1 in the Spectrum measurement.

READ:SPURious:CARRier:POWer? (Query Only)

Returns the carrier power in the Spurious measurement.

Conditions Measurement views: Spurious

Group Read commands

Syntax `READ:SPURious:CARRier:POWer?`

Arguments None

Returns <NRf> Carrier power in dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `READ:SPURIOUS:CARRIER:POWER?` might return `4.227`, indicating that the carrier power is 4.227 dBm.

READ:SPURious:COUNt? (Query Only)

Returns the number of spurious signals in the Spurious measurement.

Conditions Measurement views: Spurious

Group Read commands

Syntax `READ:SPURious:COUNt?`

Arguments None

Returns <NRf> The spurious count.

Examples `READ:SPURIOUS:COUNT?` might return `4`, indicating that the spurious count is 4.

READ:SPURious:PASS? (Query Only)

Returns the pass/fail limit test result in the Spurious measurement.

Conditions Measurement views: Spurious

Group Read commands

Syntax `READ:SPURious:PASS?`

Arguments None

Returns `0` (fail) or `1` (pass).

Examples `READ:SPURIOUS:PASS?` might return `1`, indicating that the limit test was successful.

READ:SPURious:SPECtrum:X? (Query Only)

Returns the frequencies of the spectrum trace in the Spurious measurement.

Conditions Measurement views: Spurious

Group Read commands

Syntax `READ:SPURious:SPECtrum:X?`

Arguments None

Returns `#<num_digit><num_byte><x(1)><x(2)>...<x(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<x(n)>` is the frequency (Hz) at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:SPURIOUS:SPECTRUM:X?` might return `#516020xxxx...` (16020-byte data) for the frequencies of the spectrum trace in the Spurious measurement.

READ:SPURious:SPECtrum:XY? (Query Only)

Returns the frequency and amplitude pairs of the spectrum trace in the Spurious measurement.

Conditions Measurement views: Spurious

Group Read commands

Syntax `READ:SPURious:SPECtrum:XY?`

Arguments None

Returns `#<num_digit><num_byte><x(1)><y(1)><x(2)><y(2)>...<x(n)><y(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<x(n)><y(n)>` is the frequency (Hz) and amplitude (dBm) pair at the n^{th} data point,
4-byte little endian floating-point format specified in IEEE 488.2.
The amplitude unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `READ:SPURIOUS:SPECTRUM:XY?` might return `#516020xxxx...` (16020-byte data) for the frequency and amplitude pairs of the spectrum trace in the Spurious measurement.

READ:SPURious:SPECtrum[:Y]? (Query Only)

Returns the amplitudes of the spectrum trace in the Spurious measurement.

Conditions Measurement views: Spurious

Group Read commands

Syntax `READ:SPURious:SPECtrum[:Y]?`

Related Commands READ:SPURious:SPECtrum:X?

Arguments None

Returns `#<num_digit><num_byte><y(1)><y(2)>...<y(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.

<y(n)> is the amplitude (dBm) at the nth data point,
4-byte little endian floating-point format specified in IEEE 488.2.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples READ:SPURIOUS:SPECTRUM:Y might return #516020xxxx... (16020-byte data) for the amplitudes of the spectrum trace in the Spurious measurement.

READ:SPURious:SPUR<x>:AMPLitude:ABSolute? (Query Only)

Returns the absolute amplitude of the specified spurious signal in the Spurious measurement.

Conditions Measurement views: Spurious

Group Read commands

Syntax READ:SPURious:SPUR<x>:AMPLitude:ABSolute?

Arguments None

Returns <NRf> Absolute amplitude of the specified spurious in dBm.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples READ:SPURIOUS:SPUR1:AMPLITUDE:ABSOLUTE? might return -19.782, indicating that the absolute amplitude of Spurious #1 is -19.782 dBm.

READ:SPURious:SPUR<x>:AMPLitude:RELative? (Query Only)

Returns the relative amplitude of the specified spurious signal in the Spurious measurement.

Conditions Measurement views: Spurious

Group Read commands

Syntax READ:SPURious:SPUR<x>:AMPLitude:RELative?

Arguments None

Returns <NRf> Relative amplitude of the specified spurious signal in dB. Use the [SENSe]:SPURious:REFerence command to set the power reference.

Examples READ:SPURIOUS:SPUR1:AMPLITUDE:RELATIVE? might return -9.782, indicating that the relative amplitude of Spurious #1 is -9.782 dB.

READ:SPURious:SPUR<x>:FREQuency:ABSolute? (Query Only)

Returns the absolute frequency of the specified spurious signal in the Spurious measurement.

Conditions Measurement views: Spurious

Group Read commands

Syntax READ:SPURious:SPUR<x>:FREQuency:ABSolute?

Arguments None

Returns <NRf> Absolute frequency of the spurious signal in Hz.

Examples READ:SPURIOUS:SPUR1:FREQUENCY:ABSOLUTE? might return 2.235E+9, indicating that the absolute frequency of Spurious #1 is 2.235 GHz.

READ:SPURious:SPUR<x>:FREQuency:RELative? (Query Only)

Returns the relative frequency of the specified spurious signal to the carrier in the Spurious measurement. This command is valid when [SENSe]:SPURious:REFerence is set to CARRier.

Conditions Measurement views: Spurious

Group Read commands

Syntax READ:SPURious:SPUR<x>:FREQuency:RELative?

Arguments None

Returns <NRf> Relative frequency of the spurious signal to the carrier in Hz.

Examples READ:SPURIOUS:SPUR1:FREQUENCY:RELATIVE? might return 3.634E+6, indicating that the relative frequency of Spurious #1 is 3.634 MHz.

READ:SPURious:SPUR<x>:LIMit:ABSolute? (Query Only)

Returns the absolute amplitude of the limit for the specified spurious signal in the Spurious measurement.

Conditions Measurement views: Spurious

Group Read commands

Syntax READ:SPURious:SPUR<x>:LIMit:ABSolute?

Arguments None

Returns <NRf> Absolute amplitude of the limit for the specified spurious signal in dBm. The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples READ:SPURIOUS:SPUR1:LIMIT:ABSOLUTE? might return -50.0, indicating that the absolute amplitude of the limit for Spurious #1 is -50 dBm.

READ:SPURious:SPUR<x>:LIMit:RELative? (Query Only)

Returns the relative amplitude of the limit for the specified spurious signal in the Spurious measurement.

Conditions Measurement views: Spurious

Group Read commands

Syntax READ:SPURious:SPUR<x>:LIMit:RELative?

Arguments None

Returns <NRf> Relative amplitude of the limit for the specified spurious signal in dB. Use the [SENSe]:SPURious:REFerence command to set the power reference.

Examples READ:SPURIOUS:SPUR1:LIMIT:RELATIVE? might return -10.0, indicating that the relative amplitude of the limit for Spurious #1 is -10 dB.

READ:SPURious:SPUR<x>:LIMit:VIOLation? (Query Only)

Returns whether the specified spurious signal exceeds the limit or not.

Conditions Measurement views: Spurious

Group Read commands

Syntax READ:SPURious:SPUR<x>:LIMit:VIOLation?

Arguments None

Returns 0 (under the limit) or 1 (over the limit).

Examples READ:SPURIOUS:SPUR1:LIMIT:VIOLATION? might return 1, indicating that Spurious #1 exceeds the limit.

READ:SPURious:SPUR<x>:RANGe? (Query Only)

Returns the frequency range in which the specified spurious signal occurred.

Conditions Measurement views: Spurious

Group Read commands

Syntax READ:SPURious:SPUR<x>:RANGe?

Arguments None

Returns `<string>` "A" to "T" representing Range A to T, respectively.

Examples `READ:SPURIOUS:SPUR1:RANGE?` might return `"E"`, indicating that Spurious #1 is in Range E.

READ:SQUality:FREQuency:DEViation? (Query Only)

Returns the frequency deviation in the signal quality measurement.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to C4FM, FSK2, FSK4, FSK8, or FSK16.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:FREQuency:DEViation?`

Arguments None

Returns `<NRf>` Frequency deviation in Hz.

Examples `READ:SQUALITY:FREQUENCY:DEVIATION?` might return `12.68E+3`, indicating the frequency deviation is 12.68 kHz.

READ:SQUality:FREQuency:DEViation:TABLe? (Query Only)

Returns the number of columns and the values in the frequency deviation table for a signal quality measurement.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to C4FM, FSK2, FSK4, FSK8, or FSK16.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:FREQuency:DEViation:TABLe?`

Arguments None

Returns <Dev_Num>,{<Freq_dev_Max>,<Freq_dev_Min>,<Freq_dev_Avg>}

Where

<Dev_Num> :: <NR1> is the number of columns in the frequency deviation table.
= 2 (2FSK), 4 (4FSK, C4FM), 8 (8FSK), or 16 (16FSK)
<Freq_dev_Max> :: <NRf> is the maximum frequency deviation in Hz.
<Freq_dev_Min> :: <NRf> is the minimum frequency deviation in Hz.
<Freq_dev_Avg> :: <NRf> is the average frequency deviation in Hz.

The dataset <Freq_dev_Max>,<Freq_dev_Min>,<Freq_dev_Avg> is returned for each symbol in ascending order of its level (for example, in order of symbol -3, -1, +1, and +3 for 4FSK).

Examples READ:SQUALITY:FREQUENCY:DEVIATION:TABLE? might return 2,1.257E+3,1.039E+3,1.162E+3,1.586E+3,1.298E+3,1.425E+3 for the frequency signal, populating the results table as follows.

Deviations	-1	+1
Maximum	1.257 kHz	1.586 kHz
Minimum	1.039 kHz	1.298 kHz
Average	1.162 kHz	1.425 kHz

READ:SQUality:FREQuency:ERRor? (Query Only)

Returns the frequency error in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Read commands

Syntax READ:SQUality:FREQuency:ERRor?

Arguments None

Returns <NRf> Frequency error in Hz.

Examples `READ:SQUALITY:FREQUENCY:ERROR?` might return `612.043E+3`, indicating that the frequency error is 612.043 kHz.

READ:SQUality:GAIN:IMBalance? (Query Only)

Returns the gain imbalance in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:GAIN:IMBalance?`

Arguments None

Returns `<NRf>` Gain imbalance in dB.

Examples `READ:SQUALITY:GAIN:IMBALANCE?` might return `-57.746E-3`, indicating that the gain imbalance is -0.057746 dB.

READ:SQUality:ORIGin:OFFSet? (Query Only)

Returns the origin offset in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:ORIGin:OFFSet?`

Arguments None

Returns `<NRf>` Origin offset in dB.

Examples `READ:SQUALITY:ORIGIN:OFFSET?` might return `-44.968`, indicating that the origin offset is -44.968 dB.

READ:SQUality:PEAK:EVM? (Query Only)

Returns the peak EVM (%) in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:PEAK:EVM?`

Arguments None

Returns `<NRf>` Peak EVM in percent (%).

Examples `READ:SQUALITY:PEAK:EVM?` might return `4.276`, indicating that the peak EVM is 4.276%.

READ:SQUality:PEAK:EVM:DB? (Query Only)

Returns the peak EVM (dB) in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:PEAK:EVM:DB?`

Arguments None

Returns `<NRf>` Peak EVM in dB.

Examples `READ:SQUALITY:PEAK:EVM:DB?` might return `-27.358`, indicating that the peak EVM is -27.358 dB.

READ:SQUality:PEAK:EVM:DB:OFFSet? (Query Only)

Returns the peak offset EVM (dB) in the signal quality measurement.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to OQPSK or SOQPSK.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:PEAK:EVM:DB:OFFSet?`

Arguments None

Returns `<NRf>` Peak offset EVM in dB.

Examples `READ:SQUALITY:PEAK:EVM:DB:OFFSET?` might return `-37.624`, indicating the peak offset EVM is -37.624 dB.

READ:SQUality:PEAK:EVM:LOCation? (Query Only)

Returns the time at which the EVM is peak.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:PEAK:EVM:LOCation?`

Arguments None

Returns `<NRf>` The time in symbol number at which the EVM is peak.
The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples `READ:SQUALITY:PEAK:EVM:LOCATION?` might return `68.000`, indicating that the EVM is peak at symbol #68.000.

READ:SQUality:PEAK:EVM:LOCation:OFFSet? (Query Only)

Returns the time at which the offset EVM is peak.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to OQPSK or SOQPSK.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:PEAK:EVM:LOCation:OFFSet?`

Arguments None

Returns `<NRf>` The time in symbol number at which the offset EVM is peak. The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples `READ:SQUALITY:PEAK:EVM:LOCATION:OFFSET?` might return `123.00`, indicating that the offset EVM is peak at symbol #123.

READ:SQUality:PEAK:EVM:OFFSet? (Query Only)

Returns the peak offset EVM (%) in the signal quality measurement.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to OQPSK or SOQPSK.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:PEAK:EVM:OFFSet?`

Arguments None

Returns `<NRf>` Peak offset EVM in percent (%).

Examples `READ:SQUALITY:PEAK:EVM:OFFSET?` might return `1.298`, indicating the peak offset EVM is 1.298%.

READ:SQUality:PEAK:FERRor? (Query Only)

Returns the peak FSK error in the signal quality measurement.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to FSK2, FSK4, FSK8, or FSK16.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:PEAK:FERRor?`

Arguments None

Returns `<NRf>` Peak FSK error in percent (%).

Examples `READ:SQUALITY:PEAK:FERROR?` might return `9.136`, indicating the peak FSK error is 9.136%.

READ:SQUality:PEAK:MERRor? (Query Only)

Returns the peak magnitude error (%) in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:PEAK:MERRor?`

Arguments None

Returns `<NRf>` Peak magnitude error in percent (%).

Examples `READ:SQUALITY:PEAK:MERROR?` might return `3.595`, indicating that the peak magnitude error is 3.595%.

READ:SQUality:PEAK:MERRor:DB? (Query Only)

Returns the peak magnitude error (dB) in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:PEAK:MERRor:DB?`

Arguments None

Returns `<NRf>` Peak magnitude error in dB.

Examples `READ:SQUALITY:PEAK:MERROR:DB?` might return `-28.583`, indicating that the magnitude error is -28.583 dB.

READ:SQUality:PEAK:MERRor:LOCation? (Query Only)

Returns the time at which the magnitude error is peak.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:PEAK:MERRor:LOCation?`

Arguments None

Returns `<NRf>` The time in symbol number at which the magnitude error is peak. The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples `READ:SQUALITY:PEAK:MERROR:LOCATION?` might return `68.000`, indicating that the magnitude error is peak at symbol #68.

READ:SQUality:PEAK:PERRor? (Query Only)

Returns the peak phase error in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:PEAK:PERRor?`

Arguments None

Returns `<NRf>` Peak phase error in degrees.

Examples `READ:SQUALITY:PEAK:PERROR?` might return `1.907`, indicating that the peak phase error is 1.907 °.

READ:SQUality:PEAK:PERRor:LOCation? (Query Only)

Returns the time at which the phase error is peak.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:PEAK:PERRor:LOCation?`

Arguments None

Returns `<NRf>` The time in symbol number at which the phase error is peak.
The unit can be changed by the [SENSe]:DDEMod:TIMe:UNITs command.

Examples `READ:SQUALITY:PEAK:PERROR:LOCATION?` might return `68.000`, indicating that the phase error is peak at symbol #68.

READ:SQUality:QUADrature:ERRor? (Query Only)

Returns the quadrature error in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:QUADrature:ERRor?`

Arguments None

Returns `<NRf>` Quadrature error in degrees.

Examples `READ:SQUALITY:QUADRATURE:ERROR?` might return `-14.264E-3`, indicating that the quadrature error is -0.014264°.

READ:SQUality:RHO? (Query Only)

Returns the ρ (waveform quality) value in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:RHO?`

Arguments None

Returns `<NRf>` ρ value.

Examples `READ:SQUALITY:RHO?` might return `998.703E-3`, indicating that ρ is 0.998703.

READ:SQUality:RMS:EVM? (Query Only)

Returns the RMS EVM (%) in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:RMS:EVM?`

Arguments None

Returns `<NRf>` RMS EVM in percent (%).

Examples `READ:SQUALITY:RMS:EVM?` might return `2.417`, indicating that the RMS EVM is 2.417%.

READ:SQUality:RMS:EVM:DB? (Query Only)

Returns the RMS EVM (dB) in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:RMS:EVM:DB?`

Arguments None

Returns `<NRf>` RMS EVM in dB.

Examples `READ:SQUALITY:RMS:EVM:DB?` might return `-32.356`, indicating that the RMS EVM is -32.356 dB.

READ:SQUality:RMS:EVM:DB:OFFSet? (Query Only)

Returns the RMS offset EVM (dB) in the signal quality measurement.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to OQPSK or SOQPSK.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:RMS:EVM:DB:OFFSet?`

Arguments None

Returns `<NRf>` RMS offset EVM in dB.

Examples `READ:SQUALITY:RMS:EVM:DB:OFFSET?` might return `-41.276`, indicating the RMS offset EVM is -41.276 dB.

READ:SQUality:RMS:EVM:OFFSet? (Query Only)

Returns the RMS offset EVM (%) in the signal quality measurement.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to OQPSK or SOQPSK.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:RMS:EVM:OFFSet?`

Arguments None

Returns `<NRf>` RMS offset EVM in percent (%).

Examples `READ:SQUALITY:RMS:EVM:OFFSET?` might return `0.783`, indicating the RMS offset EVM is 0.783%.

READ:SQUality:RMS:FERRor? (Query Only)

Returns the RMS FSK error in the signal quality measurement.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to FSK2, FSK4, FSK8, or FSK16.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:RMS:FERRor?`

Arguments None

Returns `<NRf>` RMS FSK error in percent (%).

Examples `READ:SQUALITY:RMS:FERROR?` might return `8.67`, indicating the RMS FSK error is 8.67%.

READ:SQUality:RMS:MER:DB? (Query Only)

Returns the RMS MER (Modulation Error Ratio) in dB in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:RMS:MER:DB?`

Arguments None

Returns `<NRf>` RMS MER in dB.

Examples `READ:SQUALITY:RMS:MER:DB?` might return `27.394`, indicating that the RMS MER is 27.394 dB.

READ:SQUality:RMS:MERRor? (Query Only)

Returns the RMS magnitude error (%) in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:RMS:MERRor?`

Arguments None

Returns `<NRf>` RMS magnitude error in percent (%).

Examples `READ:SQUALITY:RMS:MERROR?` might return `1.837`, indicating that the RMS magnitude error is 1.837%.

READ:SQUality:RMS:MERRor:DB? (Query Only)

Returns the RMS magnitude error (dB) in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:RMS:MERRor:DB?`

Arguments None

Returns `<NRf>` RMS magnitude error in dB.

Examples `READ:SQUALITY:RMS:MERROR:DB?` might return `-34.706`, indicating that the magnitude error is -34.706 dB.

READ:SQUality:RMS:PERRor? (Query Only)

Returns the RMS phase error in the signal quality measurement.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:RMS:PERRor?`

Arguments None

Returns `<NRf>` RMS phase error in degrees.

Examples `READ:SQUALITY:RMS:PERROR?` might return `893.472E-3`, indicating that the RMS phase error is 0.893472 °.

READ:SQUality:SYMBol:LENGth? (Query Only)

Returns the number of analyzed symbols.

Conditions Measurement views: Signal quality

Group Read commands

Syntax `READ:SQUality:SYMBol:LENGth?`

Related Commands

Returns `<NR1>` indicates the length of the synch word in symbols.

Examples `READ:SQUALITY:SYMBOL:LENGTH?`might return `3`, indicating the length is three symbols.

READ:SQUality:SYMBol:RATE? (Query Only)

Returns the value of the calculated symbol rate in Hz.

Conditions Measurement views: Signal quality

It is valid when the modulation type is 2|4|8|16FSK and [:SENSe]:DDEMod:SYMBol:RATE:SEARch is ON.

Group Read commands

Syntax `READ:SQUality:SYMBol:RATE?`

Related Commands [SENSe]:DDEMod:SYMBol:RATE:SEARch

Returns <NRf> is the calculated symbol rate in Hz.

Examples `READ:SQUALITY:SYMBOL:RATE?` might return `95.24E+3`, indicating the calculated symbol rate is 95.24 kHz.

READ:SQUality:SYMBol:RATE:ERRor? (Query Only)

Returns the value of the symbol rate error in percent (%).

Conditions Measurement views: Signal quality

It is valid when the modulation type is 2|4|8|16FSK and [:SENSe]:DDEMod:SYMBol:RATE:SEARch is ON.

Group Read commands

Syntax `READ:SQUality:SYMBol:RATE:ERRor?`

Related Commands [SENSe]:DDEMod:SYMBol:RATE:SEARch

Returns <NRf> is the symbol error percent (%).

Examples `READ:SQUALITY:SYMBOL:RATE:ERROR?` might return `-0.002`, indicating that the symbol rate error is -0.002%.

READ:TDIagram:FERRor? (Query Only)

Returns the frequency error in the trellis diagram measurement.

Conditions Measurement views: Trellis diagram

Group Read commands

Syntax `READ:TDIagram:FERRor?`

Arguments None

Returns `<freq_error> ::` `<NRf>` is the frequency error in Hz.

Examples `READ:TDIAGRAM:FERROR?` might return `-10.7E+3`, indicating the frequency error is -10.7 kHz.

READ:TDIagram:TRACe? (Query Only)

Returns the Trellis diagram trace data.

Conditions Measurement views: Trellis diagram

Group Read commands

Syntax `READ:TDIagram:TRACe?`

Arguments None

Returns `#<num_digit><num_byte><Y(1)><X(1)><Y(2)><X(2)>...<Y(n)><X(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<Y(n)><X(n)>` is the phase in degrees and the time in symbols pair at the n^{th} data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:TDIAGRAM:TRACE?` might return `#3160xxxx...` (160-byte data) for the Trellis diagram trace.

READ:TOVerview? (Query Only)

Returns the trace data in the time overview.

Conditions Measurement views: Time overview

Group Read commands

Syntax `READ:TOVerview?`

Arguments None

Returns `#<num_digit><num_byte><data(1)><data(2)>...<data(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<data(n)>` is the trace data in dBm for the point n,
4-byte little endian floating-point format specified in IEEE 488.2.
The unit can be changed by the [SENSe]:POWer:UNITs command.

Examples `READ:TOVERVIEW?` might return `#43204xxxx...` (3204-byte data) for the trace in the time overview.

READ:WLAN:CONSte:MAGNitude? (Query Only)

Returns the magnitude data for the WLAN constellation measurement.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:CONSte:MAGNitude?`

Returns Returned data is in the form

#<num_digit><num_byte><data(1)><data(2)>..<data(n)>

where

<num_digit> is the number of digits in <num_byte>

<num_byte> is the number of bytes of data that follow

<data(n)> is the data value at the nth data point in dB, 4-byte little endian floating-point format specified in IEEE 488.2

Examples `READ:WLAN:CONS:MAGN?` might return `#510400..` (10400-byte data) for the measurement.

READ:WLAN:CONSte:PHASe? (Query Only)

Returns the phase data for the WLAN Constellation measurement.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:CONSte:PHASe?`

Returns Returned data is in the form

#<num_digit><num_byte><data(1)><data(2)>..<data(n)>

where

- <num_digit> is the number of digits in <num_byte>
- <num_byte> is the number of bytes of data that follow
- <data(n)> is the data value at the nth data point in dB, 4-byte little endian floating-point format specified in IEEE 488.2

Examples `READ:WLAN:CONS:PHAS?` might return `#510400..` (10400-byte data) for the measurement.

READ:WLAN:CONSte:TYPE? (Query Only)

Returns the type (context) of the WLAN Constellation measurement.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:CONSte:TYPE?`

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in 16-bit little endian integers. The integers must be translated to the proper context as listed below.

Integer value Type (context):

- 0 Pilot
- 1 Data
- 2 Unused or Null

Examples `READ:WLAN:CONS:TYPE?` might return `#41352..` (1352-byte data) for the measurement. The actual data must be decoded to the context type using the table above.

READ:WLAN:CONSte:VALue? (Query Only)

Returns the value (decoded symbol) for the WLAN Constellation measurement.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:CONSte:VALue?`

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in 16-bit little endian integers.

Examples `READ:WLAN:CONSTE:VALUE?` might return `#41352` indicating 1352 bytes of data for the measurement.

READ:WLAN:CRESponse:MAGNitude? (Query Only)

Returns the magnitude data for the WLAN Channel Response measurement.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:CRESponse:MAGNitude?`

Returns Returned data is in the form

#<num_digit><num_byte><data(1)><data(2)>..<data(n)>

where

<num_digit> is the number of digits in <num_byte>

<num_byte> is the number of bytes of data that follow

<data(n)> is the data value at the nth data point in dB, 4-byte little endian floating-point format specified in IEEE 488.2

Examples `READ:WLAN:CRES:MAGN?` might return `#510400..` (10400-byte data) for the measurement.

READ:WLAN:CRESponse:PHASe? (Query Only)

Returns the phase data for the WLAN Channel Response measurement.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:CRESponse:PHASe?`

Returns Returned data is in the form

#<num_digit><num_byte><data(1)><data(2)>..<data(n)>

where

- <num_digit> is the number of digits in <num_byte>
- <num_byte> is the number of bytes of data that follow
- <data(n)> is the data value at the nth data point in dB, 4-byte little endian floating-point format specified in IEEE 488.2

Examples `READ:WLAN:CRES:PHAS?` might return `#510400` (10400-byte data) representing the phase data for the measurement.

READ:WLAN:EVM:TRACe<x>? (Query Only)

Returns the trace data for the selected WLAN EVM trace.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbol.

When <x> is 3, the parameter is Average versus Subcarrier

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax READ:WLAN:EVM:TRACe<x>?

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where:

<num_digit> is the number of digits in <num_byte.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples READ:WLAN:EVM:TRAC2? might return #510400.. (10400-byte data) for the measurement.

READ:WLAN:FLATness:PASS? (Query Only)

Queries whether the average power level values of the subcarriers across the signal bandwidth on the WLAN Spectral Flatness display remain within the limits defined for a particular standard.

NOTE. *WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.*

Group Read commands

Syntax READ:WLAN:FLATness:PASS?

Returns Returns "1" in the case that all data remains within the defined limits, and "0" otherwise.

Examples READ:WLAN:FLAT:PASS? might return 1, which means that the average power level values of subcarriers across the signal bandwidth remains within the defined limits.

READ:WLAN:FLATness:RESult? (Query Only)

Returns the summary results of each segment of the WLAN Spectral Flatness display.

NOTE. *WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:FLATness:RESult?`

Returns The data is returned as 12 comma delineated text values. The specific values are :

- minimum subcarrier within segment 1
- minimum subcarrier average energy deviation from the global average within segment 1
- deviation the of minimum subcarrier average from the allowed value in segment 1
- maximum subcarrier within segment 1
- maximum subcarrier average energy deviation from the global average within segment 1
- deviation of the maximum subcarrier average from the allowed value in segment 1
- minimum subcarrier within segment 2
- minimum subcarrier average energy deviation from the global average within segment 2
- deviation the of minimum subcarrier average from the allowed value in segment 2
- maximum subcarrier within segment 2
- maximum subcarrier average energy deviation from the global average within segment 2
- deviation of the maximum subcarrier average from the allowed value in segment 2

Examples `READ:WLAN:FLAT:RES?` might return
`-6.0000000000,-17.0626174659E-3,3.9829373825,12.0000000000,19.979260`
which represents the summary results of each segment of the WLAN Spectral Flatness display.

READ:WLAN:FLATness:TRACe<x>? (Query Only)

Returns the WLAN Spectral Flatness trace data. When <x> is 1, the parameter is Matrix (symbol deviation per subcarrier). When<x> is 2, the parameter is Average Deviation vs Subcarrier.

NOTE. *WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:FLATness:TRACe<x>?`

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where:

<num_digit> is the number of digits in <num_byte.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:WLAN:FLAT:TRAC2` might return `#510400..` (10400-byte data) for the measurement.

READ:WLAN:MERRor:TRACe<x>? (Query Only)

Returns the trace data for the selected WLAN Magnitude Error trace.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbol.

When <x> is 3, the parameter is Average versus Subcarrier

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:MERRor:TRACe<x>?`

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where:

<num_digit> is the number of digits in <num_byte.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:WLAN:MERR:TRAC2?` might return `#510400..` (10400-byte data) for the measurement.

READ:WLAN:PERRor:TRACe<x>? (Query Only)

Returns the trace data for the selected WLAN Phase Error trace.

When <x> is 1, the parameter is Matrix.

When <x> is 2, the parameter is Average versus Symbol.

When <x> is 3, the parameter is Average versus Subcarrier

Group Read commands

Syntax `READ:WLAN:PERRor:TRACe<x>?`

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where:

<num_digit> is the number of digits in <num_byte.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point in percent (%), 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:WLAN:PERR:TRAC2?` might return `#510400..` (10400-byte data) for the measurement.

READ:WLAN:PVTime:BURSt:POWer? (Query Only)

Returns the average power of the burst packet, in dBm, for the WLAN Power vs. Time trace.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:PVTime:BURSt:POWer?`

Returns Floating point number that represents the average power of the burst packet, in dBm.

Examples `FETCH:WLAN:PVT:BURST:POWER?` might return `-1.99`, which represents the value of the average power of the burst packet of -1.99 dBm.

READ:WLAN:PVTime:BURSt:WIDTh? (Query Only)

Returns the value of the measured time width of the burst packet from Power-On to Power-Down (or end of waveform, if that occurs before Power-Down), in seconds, for the WLAN Power vs. Time trace.

Group Read commands

Syntax `READ:WLAN:PVTime:BURSt:WIDTh?`

Returns Floating point number which represents the value of the measured time width of the burst packet from Power-On to Power-Down (or end of waveform, if that occurs before Power-Down), in seconds.

Examples `READ:WLAN:PVT:BURST:WIDTH?` might return `156.0000000000E-6`, which represents a burst packet width of 156 µS.

READ:WLAN:PVTime:FALL:TIME? (Query Only)

Queries the value of the width of the falling edge, in seconds.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:PVTime:FALL:TIME?`

Returns Floating point number which represents the value of the width of the falling edge of the burst, in seconds.

Examples `READ:WLAN:PVT:FALL:TIME?` might return `7e-6`, which indicates the width value of the falling edge is 7 µs.

READ:WLAN:PVTime:FALL:TIME:STARt? (Query Only)

Queries the value of the start time of the falling edge, in seconds.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:PVTime:FALL:TIME:STARt?`

Returns Floating point number which represents the value of the start time of the falling edge of the burst, in seconds.

Examples `READ:WLAN:PVT:FALL:TIME:STAR?` might return `6e-6`, which indicates that the start time value of the falling edge is at 6 µs.

READ:WLAN:PVTime:FALL:TIME:STOP? (Query Only)

Queries the value of the stop time of the falling edge, in seconds.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:PVTime:FALL:TIME:STOP?`

Returns Floating point number which represents the value of the stop time of the falling edge of the burst, in seconds.

Examples `READ:WLAN:PVT:FALL:TIME:STOP?` might return `7e-6`, which indicates the stop time value of the falling edge is at 7 µs.

READ:WLAN:PVTime:FTIMe? (Query Only)

Queries the value of the width of the falling edge, in seconds.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:PVTime:FTIMe?`

Returns Floating point number which represents the value of the width of the falling edge of the burst, in seconds.

Examples `READ:WLAN:PVT:FALL:TIME?` might return `7e-6`, which indicates the width of the falling edge is 7 µs.

READ:WLAN:PVTime:RISE:TIME? (Query Only)

Queries the value of the width of the rising edge, in seconds.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:PVTime:RISE:TIME?`

Returns Floating point number which represents the value of the width of the rising edge of the burst, in seconds.

Examples `FETCH:WLAN:PVT:RISE:TIME?` might return `7e-6`, which indicates the width value of the rising edge is 7 µs.

READ:WLAN:PVTime:RISE:TIME:STARt? (Query Only)

Queries the value of the start time of the rising edge, in seconds.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:PVTime:RISE:TIME:STARt?`

Returns Floating point number which represents the value of the start time of the rising edge of the burst, in seconds.

Examples `READ:WLAN:PVT:RISE:TIME:STAR?` might return `6e-6`, which indicates the start time value of the rising edge is at 6 µs.

READ:WLAN:PVTime:RISE:TIME:STOP? (Query Only)

Queries the value of the stop time of the rising edge, in seconds.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:PVTime:RISE:TIME:STOP?`

Returns Floating point number which represents the value of the stop time of the rising edge of the burst, in seconds.

Examples `FETCH:WLAN:PVT:RISE:TIME:STOP?` might return `7e-6`, which indicates the stop time value of the rising edge is at 7µs.

READ:WLAN:PVTime:RTIMe? (Query Only)

Queries the value of the width of the rising edge, in seconds.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:PVTime:RTIMe?`

Returns Floating point number which represents the value of the width of the rising edge of the burst, in seconds.

Examples `READ:WLAN:PVT:RISE:TIME?` might return `7e-6`, which indicates the width value of the rising edge is 7 µs.

READ:WLAN:PVTime:TRACe:X? (Query Only)

Returns the horizontal values (time in seconds) for the WLAN Power vs. Time trace.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:PVTime:TRACe:X?`

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where:

<num_digit> the number of digits in <num_byte.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:WLAN:PVT:TRAC:X` might return `#43204` (3204-byte data), which represents the value of the time in seconds (horizontal values) for the WLAN Power vs. Time trace.

READ:WLAN:PVTime:TRACe:XY? (Query Only)

Returns the horizontal value (time in seconds) and vertical value (power) for the WLAN Power vs. Time trace.

<x(n)><y(n)> is the horizontal value and vertical value pair at the nth data point.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:PVTime:TRACe:XY?`

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where:

<num_digit> is the number of digits in <num_byte.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `READ:WLAN:PVT:TRAC:XY?` might return `#43204` (3204-byte data), which represents the horizontal value (time in seconds) and vertical value (power) pair at the nth data point.

READ:WLAN:PVTime:TRACe[:Y]? (Query Only)

Returns the vertical values (power) for the WLAN Power vs. Time trace.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:PVTime:TRACe[:Y]?`

Returns #<num_digit><num_byte><data(1)><data(2)>..<data(n)>

Where:

<num_digit> is the number of digits in <num_byte.

<num_byte> is the number of bytes of data that follow.

<data(n)> is the data value at the nth data point, 4-byte little endian floating-point format specified in IEEE 488.2.

Examples `FETCH:WLAN:PVT:TRAC:Y?` might return `#43204` (3204-byte data), which represents the vertical values (power) for the WLAN Power vs. Time trace.

READ:WLAN:STABle:VALUe? (Query Only)

Returns the number of bytes in the symbol table for the WLAN measurement.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:STABle:VALUe?`

Returns #<num_digit><num_byte>

Where

<num_digit> is the number of digits in <num_byte>.

<num_byte> is the number of bytes of data that follow.

Examples `READ:WLAN:STABLE:VALUE?` might return `#43848` indicating that there are 3848 bytes of data in the symbol table.

READ:WLAN:SUMMary:BURSt:INDex? (Query Only)

Returns the index of the analyzed packet within the analysis record.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:BURSt:INDex?`

Arguments None

Returns <NRf> data in dB.

READ:WLAN:SUMMary:BURSt:POWer? (Query Only)

Returns the average power of all symbols in the packet, including Preamble and Data segments.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:BURSt:POWer?`

Arguments None

Returns <NRf> data in dBm.

READ:WLAN:SUMMary:BURSt:POWer:CFACtor? (Query Only)

Returns the peak-to-average burst power factor in the WLAN measurement. The ratio of the highest instantaneous signal power level to the average signal power.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:BURSt:POWer:CFACtor?`

Arguments None

Returns <NRf> data in dB.

READ:WLAN:SUMMary:CERRor? (Query Only)

Returns the symbol clock error in the WLAN measurement.

Not available for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:CERRor?`

Arguments None

Returns <NRf> data in parts per million.

READ:WLAN:SUMMary:CPE? (Query Only)

Returns the RMS magnitude error of the pilots over all data symbols.

Not available for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:CPE?`

Arguments None

Returns <NRf> data in percent.

READ:WLAN:SUMMary:EVM:PEAK:ALL? (Query Only)

Returns the peak EVM value for all subcarriers in the WLAN measurement or for all chips depending on the selected standard.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:EVM:PEAK:ALL?`

Arguments None

Returns <NRf> data in dB.

READ:WLAN:SUMMary:EVM:PEAK:CHIP:ALL? (Query Only)

Returns the chip at which the reported peak EVM value occurred.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:EVM:PEAK:CHIP:ALL?`

Arguments None

Returns <NR1>

READ:WLAN:SUMMary:EVM:PEAK:CHIP:ONEKchips? (Query Only)

Returns the chip at which the reported peak EVM value over the first 1000 chips occurred.

Only available for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:EVM:PEAK:CHIP:ONEKchips?`

Arguments None

Returns <NR1>

READ:WLAN:SUMMary:EVM:PEAK:DATA? (Query Only)

Returns the peak EVM value for the data subcarriers in the WLAN measurement.

Not available for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:EVM:PEAK:DATA?`

Arguments None

Returns <NRf> data in dB.

READ:WLAN:SUMMary:EVM:PEAK:ONEKchips? (Query Only)

Returns the peak EVM value measured over the first 1000 chips.

Available only for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:EVM:PEAK:ONEKchips?`

Arguments None

Returns <NRf> data in dB.

READ:WLAN:SUMMary:EVM:PEAK:PILots? (Query Only)

Returns the peak EVM value for the pilots in the WLAN measurement.

Not available for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:EVM:PEAK:PILots?`

Arguments None

Returns <NRf> data in dB.

READ:WLAN:SUMMary:EVM:PEAK:SCARrier:ALL? (Query Only)

Returns the subcarrier on which the ALL peak EVM value occurred.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:EVM:PEAK:SCARrier:ALL?`

Arguments None

Returns <NR1>

READ:WLAN:SUMMary:EVM:PEAK:SCARrier:DATA? (Query Only)

Returns the subcarrier on which the DATA peak EVM value occurred.

Not available for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:EVM:PEAK:SCARrier:DATA?`

Arguments None

Returns <NR1>

READ:WLAN:SUMMary:EVM:PEAK:SCARrier:PILots? (Query Only)

Returns the subcarrier on which the PILOT peak EVM value occurred.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:EVM:PEAK:SCARrier:PILots?`

Arguments None

Returns <NR1>

READ:WLAN:SUMMary:EVM:PEAK:SYMBol:ALL? (Query Only)

Returns the symbol on which the reported EVM value occurred.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:EVM:PEAK:SYMBol:ALL?`

Arguments None

Returns <NR1>

READ:WLAN:SUMMary:EVM:PEAK:SYMBol:DATA? (Query Only)

Returns the symbol on which the DATA peak EVM value occurred.

Not available for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:EVM:PEAK:SYMBol:DATA?`

Arguments None

Returns <NR1>

READ:WLAN:SUMMary:EVM:PEAK:SYMBol:PILots? (Query Only)

Returns the symbol on which the PILOT peak EVM value occurred.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:EVM:PEAK:SYMBol:PILots?`

Arguments None

Returns <NR1>

READ:WLAN:SUMMary:EVM:RMS:ALL? (Query Only)

Returns the RMS EVM data for all subcarriers in the WLAN measurement and for all chips for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:EVM:RMS:ALL?`

Arguments None

Returns <NRf> RMS EVM in dB

READ:WLAN:SUMMary:EVM:RMS:AVERage? (Query Only)

Returns the average RMS EVM value in the WLAN measurement over the selected number of bursts.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:EVM:RMS:AVERage?`

Arguments None

Returns <NRf> RMS EVM in dB

READ:WLAN:SUMMary:EVM:RMS:DATA? (Query Only)

Returns the RMS EVM data for the data subcarriers in the WLAN measurement.

Not available for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:EVM:RMS:DATA?`

Arguments None

Returns <NRf> RMS EVM in dB

READ:WLAN:SUMMary:EVM:RMS:MAXimum? (Query Only)

Returns the maximum RMS EVM value in the WLAN measurement over the selected number of bursts.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:EVM:RMS:MAXimum?`

Arguments None

Returns <NRf> RMS EVM in dB

READ:WLAN:SUMMary:EVM:RMS:ONEKchips? (Query Only)

Returns the RMS EVM data for the 1k chips in the WLAN measurement measured over the first 1000 chips.

Only available for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:EVM:RMS:ONEKchips?`

Arguments None

Returns <NRf> RMS EVM in dB

READ:WLAN:SUMMary:EVM:RMS:PILots? (Query Only)

Returns the RMS EVM data for the pilot subcarriers in the WLAN measurement.

Not available for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:EVM:RMS:PILots?`

Arguments None

Returns <NRf> data in percent.

READ:WLAN:SUMMary:FERRor? (Query Only)

Returns the frequency difference between the measured carrier frequency of the signal and the measurement frequency setting.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:FERRor?`

Arguments None

Returns <NRf> data in kHz.

READ:WLAN:SUMMary:HEADer? (Query Only)

Returns the received packet header information for the WLAN measurement. Applies to the 802.11b standard only.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:HEADer?`

Arguments None

Returns Results are returned in the same left-to-right order in which they are presented in the display.

READ:WLAN:SUMMary:HTSig? (Query Only)

Returns the decimal values of the received packet HT-SIGNAL. Applies to the 802.11n standard only.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:HTSig?`

Arguments None

Returns Results are returned in the same left-to-right order in which they are presented in the display.

READ:WLAN:SUMMary:IQOFfset? (Query Only)

Returns the average magnitude of the DC subcarrier level relative to total signal power.

Group Read commands

Syntax `READ:WLAN:SUMMary:IQOFfset?`

Arguments None

Returns <NRf> data in dB.

READ:WLAN:SUMMary:LSIG? (Query Only)

Returns the decimal values of the received packet (legacy) SIGNAL symbols fields. Applies to the 802.11n standard only.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:LSIG?`

Arguments None

Returns Results are returned in the same left-to-right order in which they are presented in the display.

READ:WLAN:SUMMary:PACKet? (Query Only)

Returns the type and number of symbols, EVM-RMS and average power of the Preamble and Data portions of the packet.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:PACKet?`

Arguments None

Returns Results are returned in the same left-to-right order in which they are presented in the display.

READ:WLAN:SUMMary:PACKet:DMODulation? (Query Only)

Returns the modulation used in the Data symbols.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:PACKet:DMODulation?`

Arguments None

Returns <string> The data modulation being used.

READ:WLAN:SUMMary:PACKet:FORMat? (Query Only)

Returns the packet format setting for the WLAN measurement

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:PACKet:FORMat?`

Arguments None

Returns <string> The packet format setting.

READ:WLAN:SUMMary:PACKet:GUARd? (Query Only)

Returns the packet guard interval used by the DATA symbols.

Not available for 802.11b.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:PACKet:GUARd?`

Arguments None

Returns <NRf> The packet guard interval, 1/4 or 1/8.

READ:WLAN:SUMMary:SIG? (Query Only)

Returns the decimal values of the received packet SIGNAL symbols fields.

Available only for 802.11agjp.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:SIG?`

Arguments None

Returns Results are returned in the same left-to-right order in which they are presented in the display.

READ:WLAN:SUMMary:VHTSig:A? (Query Only)

Returns the decimal values of the received packet VHT-SIGNAL(A) symbols' fields.

Available only for 802.11ac.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:VHTSig:A?`

Arguments None

Returns Results are returned in the same left-to-right order in which they are presented in the display.

READ:WLAN:SUMMary:VHTSig:B? (Query Only)

Returns the decimal values of the received packet VHT-SIGNAL(B) symbols fields.

Available only for 802.11ac.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Read commands

Syntax `READ:WLAN:SUMMary:VHTSig:B?`

Arguments None

Returns Results are returned in the same left-to-right order in which they are presented in the display.

*RST (No Query Form)

Returns the instrument settings to the factory defaults.

The *RST command does not alter the following

- The state of the GPIB interface.
- The selected GPIB address of the analyzer.
- Alignment data that affect device specifications.
- The Output Queue.
- The Service Request Enable Register setting.
- The Standard Event Status Enable Register setting.
- The Power-on status clear flag setting.
- Stored settings.

NOTE. *Execution of the *RST command is not complete until all changes from resetting the instrument are completed. Following commands and queries will not be executed until these actions are completed.*

Conditions Measurement views: All

Group IEEE common commands

Syntax `*RST`

Related Commands *CLS

Arguments None

Examples `*RST` returns the instrument settings to the factory defaults.

[SENSe]:ACPower:AVERage

Sets or queries how to average waveform in the Channel power and ACPR measurement.

Conditions Measurement views: Channel power and ACPR

Group Sense commands

Syntax

```
[SENSe]:ACPower:AVERage { OFF | TIMe | FREQuency }
[SENSe]:ACPower:AVERage?
```

Arguments `OFF` disables averaging.

`TIMe` averages waveform using time samples.

`FREQuency` averages waveform using frequency samples.

Examples `SENSE:ACPOWER:AVERAGE TIMe` averages waveform using time samples.

[SENSe]:ACPower:AVERage:COUNt

Sets or queries the number of traces for averaging in the Channel power and ACPR measurement.

Conditions Measurement views: Channel power and ACPR

Group Sense commands

Syntax

```
[SENSe]:ACPower:AVERage:COUNt <number>
[SENSe]:ACPower:AVERage:COUNt?
```

Arguments `<number> ::` `<NR1>` specifies the average count. Range: 2 to 10000.

Examples `SENSE:ACPOWER:AVERAGE:COUNT 64` sets the average count to 64.

[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution]

Sets or queries the resolution bandwidth (RBW). Programming a specified RBW sets [SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution]:AUTO OFF.

Conditions Measurement views: Channel power and ACPR

Group Sense commands

Syntax

```
[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution] <value>
[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution]?
```

Related Commands [SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution]:AUTO

Arguments `<value> ::` `<NRf>` specifies the RBW. Range: 100 Hz to 5 MHz.

Examples `SENSE:ACPOWER:BANDWIDTH:RESOLUTION 200kHz` sets the RBW to 200 kHz.

[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution]:ACTual? (Query Only)

Queries the actual resolution bandwidth (RBW) in the Channel power and ACPR measurement.

Conditions Measurement views: Channel power and ACPR

Group Sense commands

Syntax

```
[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution]:ACTual?
```

Arguments None

Returns `<NRf>` The actual RBW in Hz.

Examples `SENSE:ACPOWER:BANDWIDTH:RESOLUTION:ACTUAL?` might return `299.624E+3`, indicating that the actual RBW is 299.624 kHz.

[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution]:AUTO

Determines whether to set the resolution bandwidth (RBW) automatically or manually in the Channel power and ACPR measurement.

Conditions Measurement views: Channel power and ACPR

Group Sense commands

Syntax

```
[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution]:AUTO { OFF | ON | 0 | 1 }
[SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution]:AUTO?
```

Arguments `OFF` or `0` specifies that the RBW is set manually using the [SENSe]:ACPower:{BANDwidth|BWIDth}[:RESolution] command.

`ON` or `1` specifies that the RBW is set automatically.

Examples `SENSE:ACPOWER:BANDWIDTH:AUTO ON` sets the RBW automatically.

[SENSe]:ACPower:{BANDwidth|BWIDth}:VIDeo

Sets or queries the video bandwidth (VBW). Programming a specified VBW sets [SENSe]:ACPower:{BANDwidth|BWIDth}:VIDeo:STATe OFF.

Conditions Measurement views: Channel power and ACPR

Group Sense commands

Syntax

```
[SENSe]:ACPower:{BANDwidth|BWIDth}:VIDeo <value>
[SENSe]:ACPower:{BANDwidth|BWIDth}:VIDeo?
```

Related Commands [SENSe]:ACPower:{BANDwidth|BWIDth}:VIDeo:STATe

Arguments `<value> :: <NRf>` specifies the VBW.
Range: Current RBW/10^4 (1 Hz minimum) to Current RBW.

Examples `SENSE:ACPOWER:BANDWIDTH:VIDEO 200kHz` sets the VBW to 200 kHz.

[SENSe]:ACPower:{BANDwidth|BWIDth}:VIDeo:STATe

Determines whether to enable or disable the video bandwidth (VBW) in the Channel power and ACPR measurement.

Conditions Measurement views: Channel power and ACPR

Group Sense commands

Syntax

```
[SENSe]:ACPower:{BANDwidth|BWIDth}:VIDeo:STATe { OFF | ON
| 0 | 1 }
[SENSe]:ACPower:{BANDwidth|BWIDth}:VIDeo:STATe?
```

Arguments `OFF` or `0` disables the VBW.

`ON` or `1` enables the VBW.

Examples `SENSE:ACPOWER:BANDWIDTH:VIDEO:STATE ON` enables the VBW.

[SENSe]:ACPower:CHANnel:{BANDwidth|BWIDth}

Sets or queries frequency bandwidth of each channel (all share the same value) in the Channel power and ACPR measurement.

Conditions Measurement views: Channel power and ACPR

Group Sense commands

Syntax

```
[SENSe]:ACPower:CHANnel:{BANDwidth|BWIDth} <value>
[SENSe]:ACPower:CHANnel:{BANDwidth|BWIDth}?
```

Arguments `<value> ::` `<NRf>` specifies the channel bandwidth. Range: 1 Hz to full span.

Examples `SENSE:ACPOWER:CHANNEL:BANDWIDTH 1.5MHz` sets the channel bandwidth to 1.5 MHz.

[SENSe]:ACPower:CHANnel:FILTer

Sets or queries the adjacent channel filter in the Channel power and ACPR measurement.

Conditions Measurement views: Channel power and ACPR

Group Sense commands

Syntax

```
[SENSe]:ACPower:CHANnel:FILTer { RRCosine | NONE }
[SENSe]:ACPower:CHANnel:FILTer?
```

Arguments `RRCosine` uses the Root-Raised Cosine filter.

`NONE` uses no filter.

Examples `SENSE:ACPOWER:CHANNEL:FILTER RRCosine` uses the Root-Raised Cosine filter for the Channel power and ACPR measurement.

[SENSe]:ACPower:CHANnel:PAIRs

Sets or queries the number of adjacent channel pairs (upper and lower) in the Channel power and ACPR measurement.

Conditions Measurement views: Channel power and ACPR

Group Sense commands

Syntax

```
[SENSe]:ACPower:CHANnel:PAIRs <number>
[SENSe]:ACPower:CHANnel:PAIRs?
```

Arguments `<number> ::` `<NR1>` specifies the number of adjacent pairs. Range: 0 to 50.

Examples `SENSE:ACPOWER:CHANNEL:PAIRS 5` sets five adjacent channel pairs.

[SENSe]:ACPower:CHANnel:SPACing

Sets or queries frequency difference between centers of each channel in the Channel power and ACPR measurement.

Conditions Measurement views: Channel power and ACPR

Group Sense commands

Syntax
```
[SENSe]:ACPower:CHANnel:SPACing <value>
[SENSe]:ACPower:CHANnel:SPACing?
```

Arguments <value> :: <NRf> specifies the channel-to-channel spacing.
Range: 1 Hz to 1 GHz.

Examples SENSE:ACPOWER:CHANNEL:SPACING 5MHz sets the channel-to-channel spacing to 5 MHz.

[SENSe]:ACPower:CHIPrate

Sets or queries the chip rate when [SENSe]:ACPower:CHANnel:FILTer is set to RRCosine (Root Raised Cosine).

Conditions Measurement views: Channel power and ACPR

Group Sense commands

Syntax
```
[SENSe]:ACPower:CHIPrate <value>
[SENSe]:ACPower:CHIPrate?
```

Related Commands [SENSe]:ACPower:CHANnel:FILTer

Arguments <value> :: <NRf> specifies the chip rate. Range: 100 Hz to 105 MHz.

Examples SENSE:ACPOWER:CHIPRATE 5kHz sets the chip rate to 5 kHz.

[SENSe]:ACPower:CLEar:RESults (No Query Form)

Restarts the average trace.

Conditions Measurement views: Channel power and ACPR

Group Sense commands

Syntax `[SENSe]:ACPower:CLEar:RESults`

Arguments None

Examples `SENSE:ACPOWER:CLEAR:RESULTS` restarts the average trace.

[SENSe]:ACPower:FREQuency

Sets or queries the center frequency in the Channel power and ACPR measurement.

Conditions Measurement views: Channel power and ACPR

Group Sense commands

Syntax
```
[SENSe]:ACPower:FREQuency <value>
[SENSe]:ACPower:FREQuency?
```

Arguments `<value> ::  <NRf>` specifies the center frequency.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `SENSE:ACPOWER:FREQUENCY 2.35GHz` sets the center frequency to 2.35 GHz.

[SENSe]:ACPower:FREQuency:STEP

Sets or queries the frequency step size in the Channel power and ACPR measurement. Programming a specified step size sets [SENSe]:ACPower FREQuency:STEP:AUTO OFF.

Conditions Measurement views: Channel power and ACPR

Group Sense commands

Syntax
```
[SENSe]:ACPower:FREQuency:STEP <value>
[SENSe]:ACPower:FREQuency:STEP?
```

Related Commands [SENSe]:ACPower:FREQuency:STEP:AUTO

Arguments `<value> ::  <NRf>` specifies the frequency step size.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `SENSE:ACPOWER:FREQUENCY:STEP 1kHz` sets the frequency step size to 1 kHz.

[SENSe]:ACPower:FREQuency:STEP:AUTO

Determines whether to set the frequency step size automatically or manually in the Channel power and ACPR measurement.

Conditions Measurement views: Channel power and ACPR

Group Sense commands

Syntax
```
[SENSe]:ACPower:FREQuency:STEP:AUTO { OFF | ON | 0 | 1 }
[SENSe]:ACPower:FREQuency:STEP:AUTO?
```

Arguments `OFF` or `0` specifies that the frequency step size is set manually using the [SENSe]:ACPower:FREQuency:STEP command.

`ON` or `1` specifies that the frequency step size is set automatically.

Examples `SENSE:ACPOWER:FREQUENCY:STEP:AUTO ON` specifies that the frequency step size is set automatically.

[SENSe]:ACPower:NFLoor:STATe

Determines whether to enable or disable the correction for noise floor.

Conditions Measurement views: Channel power and ACPR

Group Sense commands

Syntax

```
[SENSe]:ACPower:NFLoor:STATe { OFF | ON | 0 | 1 }
[SENSe]:ACPower:NFLoor:STATe?
```

Arguments `OFF` or `0` disables the correction for noise floor.

`ON` or `1` enables the correction for noise floor.

Examples `SENSE:ACPOWER:NFLOOR:STATE ON` enables the correction for noise floor.

[SENSe]:ACPower:POINts:COUNt

Sets or queries the number of points used for the trace in the Channel Power and ACPR measurement.

Conditions Measurement views: Channel power and ACPR

Group Sense commands

Syntax

```
[SENSe]:ACPower:POINts:COUNt { P801 | P2401 | P4001 | P8001
| P10801 }
[SENSe]:ACPower:POINts:COUNt?
```

Arguments `RTBandwidth` optimizes the measurement for real-time bandwidth.

`DRANge` optimizes the measurement for dynamic range.

Examples `SENSE:ACPOWER:POINTS:COUNT P4001` sets the number of points used in the trace of the Channel Power and ACPR display to 4001.

[SENSe]:ACPower:RRCRolloff

Sets or queries the filter parameter (roll-off ratio) for the Root Raised Cosine filter.

Conditions Measurement views: Channel power and ACPR

Group Sense commands

Syntax
```
[SENSe]:ACPower:RRCRolloff <value>
[SENSe]:ACPower:RRCRolloff?
```

Related Commands [SENSe]:ACPower:CHANnel:FILTer

Arguments <value> :: <NRf> specifies the filter parameter.
Range: 0.0001 to 1 in 0.0001 steps.

Examples SENSE:ACPOWER:RRCROLLOFF 0.3 sets the filter parameter to 0.3.

[SENSe]:ACPower:SYMBrate

Sets or queries the symbol rate when [SENSe]:ACPower:CHANnel:FILTer is set to RRCosine (Root Raised Cosine).

Conditions Measurement views: Channel power and ACPR

Group Sense commands

Syntax
```
[SENSe]:ACPower:SYMBrate <value>
[SENSe]:ACPower:SYMBrate?
```

Related Commands [SENSe]:ACPower:CHANnel:FILTer

Arguments <value> :: <NRf> specifies the symbol rate. Range: 100 Hz to 105 MHz.

Examples SENSE:ACPOWER:CHIPRATE 5kHz sets the symbol rate to 5 kHz.

[SENSe]:ACQuisition:{BANDwidth|BWIDth}

Sets or queries the acquisition bandwidth (frequency range of the acquisition) when [SENSe]:ACQuisition:MODE is set to SAMPles or LENGth.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:ACQuisition:{BANDwidth|BWIDth} <value>
[SENSe]:ACQuisition:{BANDwidth|BWIDth}?
```

Arguments <value> :: <NRf> specifies the acquisition bandwidth.
Range: 1 MHz to 40 MHz (Standard) / 110 MHz (Option 110).

Examples SENSE:ACQUISITION:BANDWIDTH 30MHz sets the acquisition bandwidth to 30 MHz.

[SENSe]:ACQuisition:FFRame:ACTual? (Query Only)

Queries the actual number of Fast Frames.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:ACQuisition:FFRame:ACTual?
```

Arguments None

Returns <NR1> The actual number of Fast Frames.

Examples SENSE:ACQUISITION:FFRAME:ACTUAL? might return 178, indicating that the actual number of Fast Frames is 178.

[SENSe]:ACQuisition:FFRame:LIMit

Sets or queries the limit number of Fast Frames.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:ACQuisition:FFRame:LIMit
[SENSe]:ACQuisition:FFRame:LIMit?
```

Arguments <value> :: <NR1> specifies the fast frame limit. Range: 1 to 65535 frames.

Examples SENSE:ACQUISITION:FFRAME:LIMIT 500 sets the Fast Frame limit to 500.

[SENSe]:ACQuisition:FFRame:STATe

Determines whether to enable or disable the Fast Frame.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:ACQuisition:FFRame:STATe { OFF | ON | 0 | 1 }
[SENSe]:ACQuisition:FFRame:STATe?
```

Arguments OFF or 0 disables the Fast Frame.

ON or 1 enables the Fast Frame.

Examples SENSE:ACQUISITION:FFRAME:STATE ON enables the Fast Frame.

[SENSe]:ACQuisition:FSAVe:ENAble

Set or queries the FastSave acquisition mode.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:ACQuisition:FSAVe:ENAble { OFF | ON | 0 | 1 }
[SENSe]:ACQuisition:FSAVe:ENAble?
```

Arguments `OFF` or `0` disables the FastSave acquisition.

`ON` or `1` enables the FastSave acquisition.

Examples `SENSE:ACQUISITION:FSAVE:ENABLE ON` enables the FastSave acquisition.

[SENSe]:ACQuisition:FSAVe:FILEs:MAXimum

Set or queries the FastSave maximum number of files saved per run.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:ACQuisition:FSAVe:FILEs:MAXimum <value>
[SENSe]:ACQuisition:FSAVe:FILEs:MAXimum?
```

Arguments `<value> :: <NR2>` specifies the maximum number of files saved per run. Range: 1 to 1,000,000 files.

NOTE. *When the FILEs:MAXimum is reached, an event is added to the event queue to indicate that saves are completed. Note that this same event is issued now when Trigger Actions Max total files is reached as well.*

Examples `SENSE:ACQUISITION:FSAVE:FILES:MAXIMUM 10` sets the maximum number of files saved per run to 10.

[SENSe]:ACQuisition:FSAVe:FORMat

Set or queries the FastSave file format.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:ACQuisition:FSAVe:FORMat { TIQ| MAT }
[SENSe]:ACQuisition:FSAVe:FORMat?
```

Arguments `TIQ` selects the TIQ file format for FastSave acquisitions.

`MAT` selects the MAT file format for FastSave acquisitions..

Examples `SENSE:ACQUISITION:FSAVE:FORMAT MAT` selects the Matlab file format.

[SENSe]:ACQuisition:FSAVe:LOCation

Set or queries the FastSave file save location (a directory).

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:ACQuisition:FSAVe:LOCation <path>
[SENSe]:ACQuisition:FSAVe:LOCation?
```

Arguments `<path> :: <string>` specifies the location where FastSave acquisitions will be saved.

Examples `SENSE:ACQUISITION:FSAVE:LOCATION "C:\RSA5100B Files\Test_folder"` sets location where FastSave acquisitions are saved to C:\RSA5100B Files\Test_folder.

[SENSe]:ACQuisition:FSAVe:NAME:BASE

Set or queries the FastSave Base file name.

Conditions Measurement views: All

Group Sense commands

Syntax

```
[SENSe]:ACQuisition:FSAVe:NAME:BASE <file_name>
[SENSe]:ACQuisition:FSAVe:NAME:BASE?
```

Arguments

`<file_name> ::  <string>` specifies the base file name use for FastSave acquisitions.

Examples

`SENSE:ACQUISITION:FSAVE:NAME:BASE "FSAVE"` sets the Base file name to FSAVE.

[SENSe]:ACQuisition:MEMory:AVAilable:SAMPles? (Query Only)

Returns the amount of acquisition memory available in the instrument.

Conditions Measurement views: All

Group Sense commands

Syntax

```
[SENSe]:ACQuisition:MEMory:AVAilable:SAMPles?
```

Arguments None

Returns `<NRf>` The amount of acquisition memory available in samples.

Examples `SENSE:ACQUISITION:MEMORY:AVAILABLE:SAMPLES?` might return `999.424E+3`, indicating that 999424 samples are available.

[SENSe]:ACQuisition:MEMory:CAPacity[:TIME]? (Query Only)

Returns the acquisition memory capacity (maximum period of time that can be acquired with the acquisition memory).

Conditions Measurement views: All

Group Sense commands

Syntax

```
[SENSe]:ACQuisition:MEMory:CAPacity[:TIME]?
```

Arguments None

Returns <NRf> The acquisition memory capacity in seconds.

Examples SENSE:ACQUISITION:MEMORY:CAPACITY:TIME? might return 26.651E-3, indicating that 26.651 ms can be acquired.

[SENSe]:ACQuisition:MEMory:USED[:PERCent]? (Query Only)

Returns the percentage of the capacity used based on the current settings.

Conditions Measurement views: All

Group Sense commands

Syntax [SENSe]:ACQuisition:MEMory:USED[:PERCent]?

Arguments None

Returns <NRf> The percentage of the capacity used.

Examples SENSE:ACQUISITION:MEMORY:USED:PERCENT? might return 50.0, indicating that 50% is used.

[SENSe]:ACQuisition:MODE

Sets or queries the acquisition mode (how to determine the sampling parameters of acquisition bandwidth, samples, and length).

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:ACQuisition:MODE { AUTO | SAMPles | LENGth }
[SENSe]:ACQuisition:MODE?
```

Related Commands [SENSe]:ACQuisition:{BANDwidth|BWIDth}, [SENSe]:ACQuisition:SAMPles, [SENSe]:ACQuisition:SEConds

Arguments `AUTO` sets the all sampling parameters automatically.

`SAMPles` sets the acquisition bandwidth and samples manually, using the [SENSe]:ACQuisition:{BANDwidth|BWIDth} and :SAMPles commands.

`LENGth` sets the acquisition bandwidth and length manually, using the [SENSe]:ACQuisition:{BANDwidth|BWIDth} and :SEConds commands.

Examples `SENSE:ACQUISITION:MODE AUTO` sets the all sampling parameters automatically.

[SENSe]:ACQuisition:OPTimization

Sets or queries the method of the global gain and input bandwidth optimization.

This command replaces the following commands:

- [:SENSe]:ACPower:OPTimize:SPAN(?)
- [:SENSe]:DPX:{BANDwidth|BWIDth}:OPTimization(?)
- [:SENSe]:SGRam:{BANDwidth|BWIDTH}:OPTimization(?)
- [:SENSe]:SPECtrum:{BANDwidth|BWIDTH}:OPTimization(?)
- [:SENSe]:MCPower:OPTimize:SPAN(?)

When a setup or tiq file that was created with an earlier version of software is recalled, the instrument selects a setting for the global optimization that will result in the recalled measurement-specific optimizations. If recalled measurements have conflicting optimizations, then the instrument selects one of the optimizations and issues a GPIB event (event number 108).

Conditions Measurement views: Spectrum, DPX, Spectrum, Spectrogram, ACPR, MCPR

Group Sense commands

Syntax

```
[SENSe]:ACQuisition:OPTimization { BEST | AUTO | MAXDynrange
| MINNoise | MINTime }
[SENSe]:ACQuisition:OPTimization?
```

Arguments

`BEST` (Best for multiple displays) causes the instrument to evaluate all the open displays and select an Acquisition Bandwidth wide enough to satisfy as many measurements as possible

`AUTO` sets the gain and input bandwidth for an optimized trade-off of the competing performance choices.

`MAXDynrange` optimizes the gain and input bandwidth to maximize the dynamic range.

`MINNoise` optimizes the gain and input bandwidth to minimize noise.

`MINTime` optimizes the gain and input bandwidth to minimize sweep time.

Examples

`SENSE:SPECTRUM:BANDWIDTH:OPTIMIZATION AUTO` optimizes automatically the gain and input bandwidth.

[SENSe]:ACQuisition:RECord:FILE:LENGth

Sets or queries how much data (time in milliseconds, seconds, or minutes) is recorded when recording starts. File Length uses the Max saved files per run: setting to determine how many data files are captured per each recording session.

Conditions

Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group

Sense group

Syntax

```
[SENSe]:ACQuisition:RECord:FILE:LENGth <numeric>
[SENSe]:ACQuisition:RECord:FILE:LENGth?
```

Arguments

<numeric>::=<NR3> determines the file length of each data file saved.

Range: 1E-21 to 10E3 when units set to milliseconds

Range: 1E-5 to 10E3 when units set to seconds

Range: 1E-13 to 10E3 when units set to minutes

Returns

Examples

`[SENSE]:ACQUISITION:RECORD:FILE:LENGTH` 10E2 sets the file length to 1000.

[SENSe]:ACQuisition:RECord:FILE:LENGth:ENABle

Sets or queries the file length conditions to determine how much data is recorded. File Length uses the Max saved files per run: setting to determine how many data files are captured per each recording session.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Sense group

Syntax
```
[SENSe]:ACQuisition:RECord:FILE:LENGth:ENABle { OFF | ON | 0 | 1 }
[SENSe]:ACQuisition:RECord:FILE:LENGth:ENABle?
```

Arguments ON or 1 enables File Length.

OFF or 0 disables File Length.

Examples `[SENSE]:ACQUISITION:RECORD:FILE:LENGTH:ENABLE` ON enables the File Length conditions.

[SENSe]:ACQuisition:RECord:FILE:LENGth:UNITs

Sets or queries the units used for the file length setting.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Sense group

Syntax
```
[SENSe]:ACQuisition:RECord:FILE:LENGth:UNITs { MSEC | SEC | MIN }
```

Arguments MSEC: milliseconds

SEC: seconds

MIN: minutes

Examples `[SENSE]:ACQUISITION:RECORD:FILE:LENGTH:UNITS MSEC` sets the units for the file length to milliseconds.

[SENSe]:ACQuisition:RECord:FILE:LOCation

Sets the file location of the saved data records. To fully support data streaming, the drive selected for recording data must have a minimum write speed of 300 MB/sec.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Sense group

Syntax

```
[SENSe]:ACQuisition:RECord:FILE:LOCation <file_path>
[SENSe]:ACQuisition:RECord:FILE:LOCation?
```

Arguments <file_path>::=<"string"> specifies the location.

If you omit the directory path, the default path is used, which is C:\SignalVu-PC Files\Sample Data Records initially.

Examples `[SENSE]:ACQUISITION:RECORD:FILE:LOCATION` C:\SignalVu-PC Files\Sample Data Records sets the location of the data records to that location.

[SENSe]:ACQuisition:RECord:FILE:MAXimum

Sets or queries how many times a record is created, of the specified File Length setting. If the File Length is not enabled, this setting is ignored. See the command [:SENSe]:ACQuisition:RECord:FILE:LENGth:ENABle.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Sense group

Syntax

```
[SENSe]:ACQuisition:RECord:FILE:MAXimum <value>
[SENSe]:ACQuisition:RECord:FILE:MAXimum?
```

Arguments <value>::=<NR1> the number of records created.

Range: 1 to 1000000.

Examples `[SENSE]:ACQUISITION:RECORD:FILE:MAXIMUM 10` sets the number of files to saved to ten.

[SENSe]:ACQuisition:RECord:FILE:NAME:BASE

Sets or queries the base file name that is used for all data recordings. The base name is appended with additional information, creating unique file names.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Sense group

Syntax

```
[SENSe]:ACQuisition:RECord:FILE:NAME:BASE <base_name>
[SENSe]:ACQuisition:RECord:FILE:NAME:BASE?
```

Arguments <base_name>::=<"string"> specifies the base file name used for all data recordings.

Examples `[SENSE]:ACQUISITION:RECORD:FILE:NAME:BASE` "DataFile" sets the base name to DataFile.

[SENSe]:ACQuisition:RECord:FILE:STRucture

Sets or queries the file structure to use for the recorded data files.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Sense group

Syntax

```
[SENSe]:ACQuisition:RECord:FILE:STRucture { RAW | FORM }
```

Arguments RAW: Unformatted files are created. Two files are created for each recording, a header file (.r3h) and the data file (.r3a).

FORMatted: Formatted files are created, ending with a .r3f suffix.

The contents of these file types is described in the RSA306 API Reference manual (077-1031-XX) available on the RSA306 Flash Drive or the Tektronix web site www.tektronix.com/downloads.

Examples `[SENSE]:ACQUISITION:RECORD:FILE:STRUCTURE FORMATTED` sets the file format to Formatted.

[SENSe]:ACQuisition:RECord:STARt (No Query Form)

Starts the data recording. The Record to disk must the set to Record now. See the command [:SENSe]:ACQuisition:RECord:TYPE. See the command [:SENSe]:ACQuisition:RECord:FILE:LOCation to specify the file name and location.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Sense group

Syntax `[SENSe]:ACQuisition:RECord:STARt`

Arguments None

Examples `[SENSE]:ACQUISITION:RECORD:START` stops the data recording.

[SENSe]:ACQuisition:RECord:TYPE

Sets or queries when the recording starts streaming data to the specified file.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Sense group

Syntax `[SENSe]:ACQuisition:RECord:TYPE { NOW | TRIGger }`

Arguments NOW starts the recording immediately when Start is enabled. See the command [:SENSe]:ACQuisition:RECord:STARt.

TRIGger starts the recording when an appropriate trigger signal is received and Triggering is set to Triggered. See the command :TRIGger[:SEQuence][:STATus].

Examples `[SENSE]:ACQUISITION:RECORD:TYPE NOW` sets the Record to disk setting to Record now.

[SENSe]:ACQuisition:SAMPles

Sets or queries the acquisition samples (number of samples acquired over the acquisition time) when [SENSe]:ACQuisition:MODE is set to SAMPles.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:ACQuisition:SAMPles <value>
[SENSe]:ACQuisition:SAMPles?
```

Arguments <value> :: <NR1> specifies the acquisition samples. Range: 2 to 1 G samples.

Examples SENSE:ACQUISITION:SAMPLES 1114 sets the acquisition samples to 1114.

[SENSe]:ACQuisition:SEConds

Sets or queries the acquisition length (time over which the acquisition occurs) when [SENSe]:ACQuisition:MODE is set to LENGth.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:ACQuisition:SEConds <value>
[SENSe]:ACQuisition:SEConds?
```

Arguments <value> :: <NRf> specifies the acquisition length.

Examples SENSE:ACQUISITION:SAMPLES 12.5ms sets the acquisition length to 12.5 ms.

[SENSe]:{AM|FM|PM}:{BANDwidth|BWIDth}:MEASurement

Sets measurement bandwidth for the AM, FM, or PM demodulation to analyze.

Conditions Measurement views: AM, FM, PM

Group Sense commands

Syntax
```
[SENSe]:{AM|FM|PM}:{BANDwidth|BWIDth}:MEASurement <value>
[SENSe]:{AM|FM|PM}:{BANDwidth|BWIDth}:MEASurement?
```

Related Commands FETCh:{AM|FM|PM}?

READ:AM:RESult?

READ:FM:RESult?

READ:PM:RESult?

Arguments `<value> ::  <NRf>` specifies the demodulation bandwidth for the specified modulation type.

Examples `SENSE:AM:BANDwidth:MEASurement 1e6` sets the amplitude demodulation bandwidth to 1 MHz.

[SENSe]:{AM|FM|PM}:CLEar:RESults (No Query Form)

Clear results in the AM/FM/PM measurement.

Conditions Measurement views: AM, FM, PM

Group Sense commands

Syntax
```
[SENSe]:{AM|FM|PM}:CLEar:RESults
```

Arguments None

Examples `SENSE:AM:CLEAR:RESULTS` clears results in the AM measurement.

[SENSe]:{AM|FM|PM}:{MTPoints|MAXTracepoints}

Sets or queries the maximum trace points in the AM/FM/PM measurement.

Conditions Measurement views: AM, FM, PM

Group Sense commands

Syntax
```
[SENSe]:{AM|FM|PM}:{MTPoints|MAXTracepoints} { ONEK | TENK |
HUNDredk | NDECimate | NEVerdecimate }
[SENSe]:{AM|FM|PM}:{MTPoints|MAXTracepoints}?
```

Arguments The following table lists the arguments.

Argument	Maximum trace points
ONEK	1k
TENK	10k
HUNDredk	100k
NDECimate or NEVerdecimate	Never decimate

Examples `SENSE:AM:MTPOINTS ONEK` selects the maximum trace points of 1000 in the AM measurement.

[SENSe]:AM:DETect:AMPLitude

Sets or queries the carrier amplitude detection method used to determine the 0% reference modulation in the AM measurement.

Conditions Measurement views: AM

Group Sense commands

Syntax
```
[SENSe]:AM:DETect:AMPLitude { AVERage | MEDian }
[SENSe]:AM:DETect:AMPLitude?
```

Arguments `AVERage` defines the 0% reference modulation as the average amplitude in the analysis range (default).

`MEDian` defines the 0% reference modulation as the median amplitude ([(maximum)+(minimum)]/2) in the analysis range.

Examples `SENSE:AM:DETECT:AMPLITUDE AVERage` defines the 0% modulation as the average amplitude in the analysis range.

[SENSe]:ANALysis:ADVanced:DITHer

Determines whether to enable or disable dithering, or set it automatically.

Dither is a random low-level signal consisting of white noise of one quantizing level peak-to-peak amplitude which may be added to an analog signal prior to sampling for the purpose of minimizing quantization error.

Conditions Measurement views: All

Group Sense commands

Syntax

```
[SENSe]:ANALysis:ADVanced:DITHer { AUTO | ON | OFF }
[SENSe]:ANALysis:ADVanced:DITHer?
```

Arguments `AUTO` specifies that the dither is set automatically.

`ON` enables dithering.

`OFF` disables dithering.

Examples `SENSE:ANALYSIS:ADVANCED:DITHER ON` enables dithering.

[SENSe]:ANALysis:ADVanced:DITHer:HWARe:STATus? (Query Only)

Returns the dithering hardware status.

Conditions Measurement views: All

Group Sense commands

Syntax

```
[SENSe]:ANALysis:ADVanced:DITHer:HWARe:STATus?
```

Arguments None

Returns One of the following status information.

Table 2-28: Dithering status

Status	Description
DUNaligned	Dithering is disabled and unaligned.
ON	Dithering is enabled
OFF	Dithering is disabled.

Examples `SENSE:ANALYSIS:ADVANCED:DITHER:HWARE:STATUS?` might return `OFF`, indicating that the dithering is disabled.

[SENSe]:ANALysis:LENGth

Sets or queries the analysis length. Programming a specified length sets [SENSe]:ANALysis:AUTO OFF.

Conditions Measurement views: All

Group Sense commands

Syntax

```
[SENSe]:ANALysis:LENGth <value>
[SENSe]:ANALysis:LENGth?
```

Related Commands [SENSe]:ANALysis:LENGth:AUTO

Arguments `<value> :: <NRf>` specifies the analysis length.
Range: 10 ns to [(acquisition length) - 400 ns].

If [(analysis start) + (analysis length)] > [(acquisition length) - 400 ns], the actual analysis length is reduced to [(acquisition length) - 200 ns].

Examples `SENSE:ANALYSIS:LENGTH 25.625us` sets the analysis length to 25.625 µs.

[SENSe]:ANALysis:LENGth:ACTual? (Query Only)

Queries the actual analysis length.

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:ANALysis:LENGth:ACTual?`

Arguments None

Returns `<NRf>` Actual analysis length in seconds.

Examples `SENSE:ANALYSIS:LENGTH:ACTUAL?` might return `25.625E-6`, indicating that the actual analysis length is 25.625 µs.

[SENSe]:ANALysis:LENGth:AUTO

Determines whether to set the analysis length automatically or manually.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:ANALysis:LENGth:AUTO { OFF | ON | 0 | 1 }
[SENSe]:ANALysis:LENGth:AUTO?
```

Arguments `OFF` or `0` sets the analysis length manually, using the [SENSe]:ANALysis:LENGth command.

`ON` or `1` sets the analysis length automatically.

Examples `SENSE:ANALYSIS:LENGTH:AUTO ON` sets the analysis length automatically.

[SENSe]:ANALysis:REFerence

Sets or queries the analysis time reference.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:ANALysis:REFerence { ACQSTART | TRIGGER }
[SENSe]:ANALysis:REFerence?
```

Arguments `ACQSTART` specifies the acquisition start as the time zero reference.

`TRIGGER` specifies the trigger point as the time zero reference.

Examples `SENSE:ANALYSIS:REFERENCE ACQSTART` specifies the acquisition start as the analysis time reference.

[SENSe]:ANALysis:STARt

Sets or queries the analysis offset time. Programming a specified offset time sets [SENSe]:ANALysis:STARt:AUTO OFF.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:ANALysis:STARt <value>
[SENSe]:ANALysis:STARt?
```

Related Commands [SENSe]:ANALysis:LENGth, [SENSe]:ANALysis:STARt:AUTO

Arguments `<value> :: <NRf>` specifies the analysis offset time.
Range: 0 to [(acquisition length) - 200 ns].

If [(analysis start) + (analysis length)] > [(acquisition length) - 400 ns], the actual analysis length is reduced to [(acquisition length) - 200 ns].

Examples `SENSE:ANALYSIS:START 23.5us` sets the analysis offset to 23.5 µs.

[SENSe]:ANALysis:STARt:AUTO

Determines whether to set the analysis offset automatically or manually.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:ANALysis:STARt:AUTO { OFF | ON | 0 | 1 }
[SENSe]:ANALysis:STARt:AUTO?
```

Arguments `OFF` or `0` sets the analysis offset manually, using the [SENSe]:ANALysis:STARt command.

`ON` or `1` sets the analysis offset automatically.

Examples `SENSE:ANALYSIS:START:AUTO ON` sets the analysis offset automatically.

[SENSe]:ANSPectrum:STARt:AUTO:METHod

Sets or queries the method used for computing auto analysis and spectrum offsets when [SENSe]:ANALysis:REFerence is set to TRIGger.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:ANSPectrum:STARt:AUTO:METHod { INCLude | LEGacy }
[SENSe]:ANSPectrum:STARt:AUTO:METHod?
```

Arguments `INCLude` includes the trigger point.

`LEGacy` starts a trigger point.

Examples `SENSE:ANSPECTRUM:START:AUTO:METHOD INCLUDE` includes the trigger point.

[SENSe]:AUDio:AVERage:COUNt

Sets or queries the number of harmonics to average in the audio measurement.

The range is from 2 to 20

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax
```
[SENSe]:AUDio:AVERage:COUNt <average_value>
[SENSe]:AUDio:AVERage:COUNt?
```

Arguments <average_value> ::= <NR1> the number of harmonics to average. .

Examples `[SENSE]:AUDIO:AVERAGE:COUNT 10` sets the number of harmonics to average to 10.

[SENSe]:AUDio:AVERage:ENABle

Enables or disables averaging of harmonics in the audio measurement.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax
```
[SENSe]:AUDio:AVERage:ENABle { OFF | ON | 0 | 1 }
[SENSe]:AUDio:AVERage:ENABle?
```

Arguments `OFF` or `0` turns averaging off.

`ON` or `1` turns averaging on.

Examples `[SENSE]:AUDIO:AVERAGE:ENABLE ON` turns on averaging of harmonics in the audio measurement.

[SENSe]:AUDio:BANDwidth|BWIDth

Sets or queries the audio measurement bandwidth.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax
```
[SENSe]:AUDio:BANDwidth|BWIDth <value>
[SENSe]:AUDio:BANDwidth|BWIDth?
```

Arguments <value> ::= <NR3> the audio measurement bandwidth value.

Examples `[SENSE]:AUDIO:BANDWIDTH|BWIDTH 20.5E+3` sets the audio measurement bandwidth to 20.5 kHz.

[SENSe]:AUDio:BANDwidth|BWIDth:RESolution

Sets or queries the resolution bandwidth for the audio measurement.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax

```
[SENSe]:AUDio:BANDwidth|BWIDth:RESolution <value>
[SENSe]:AUDio:BANDwidth|BWIDth:RESolution?
```

Arguments <value> ::= <NR3> the resolution bandwidth value.

Examples `[SENSE]:AUDIO:BANDWIDTH|BWIDTH:RESOLUTION 40` sets the resolution bandwidth to 40 Hz.

[SENSe]:AUDio:BANDwidth|BWIDth:RESolution:AUTO

Determines whether to set the resolution bandwidth frequency automatically or manually.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax

```
[SENSe]:AUDio:BANDwidth|BWIDth:RESolution:AUTO { OFF | ON
| 0 | 1 }
[SENSe]:AUDio:BANDwidth|BWIDth:RESolution:AUTO?
```

Arguments `OFF` or `0` specifies the resolution bandwidth is set manually.

`ON` or `1` specifies the resolution bandwidth is set automatically.

Examples `[SENSE]:AUDIO:BANDWIDTH|BWIDTH:RESOLUTION:AUTO ON` sets the resolution bandwidth automatically.

[SENSe]:AUDio:BANDwidth|BWIDth:RESolution:SHAPe

Sets or queries the shape of the resolution bandwidth for the audio measurement.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax
```
[SENSe]:AUDio:BANDwidth|BWIDth:RESolution:SHAPe { FLATtop
| KAISer }
[SENSe]:AUDio:BANDwidth|BWIDth:RESolution:SHAPe?
```

Arguments `FLATtop` sets the resolution bandwidth shape to Flattop.

`KAISer` sets the resolution bandwidth shape to Kaiser.

Examples `[SENSE]:AUDIO:BANDWIDTH|BWIDTH:RESOLUTION:SHAPE FLATtop` sets the resolution bandwidth to Flattop for the audio measurement.

[SENSe]:AUDio:CARRier:OFFSet

Sets or queries the carrier frequency offset. This selection is available for FM or PM measurements.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax
```
[SENSe]:AUDio:CARRier:OFFSet <value>
[SENSe]:AUDio:CARRier:OFFSet?
```

Arguments <value> ::= <NR3> the carrier frequency offset value.

Examples `[SENSE]:AUDIO:CARRIER:OFFSET 422.5` sets the carrier frequency offset to 422.5 Hz.

[SENSe]:AUDio:CARRier:OFFSet:AUTO

Determines whether to set the carrier frequency error automatically or manually. This selection is available FM or PM measurements.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax
```
[SENSe]:AUDio:CARRier:OFFSet:AUTO { OFF | ON | 0 | 1 }
[SENSe]:AUDio:CARRier:OFFSet:AUTO?
```

Arguments `OFF` or `0` specifies the carrier frequency error is set manually.

`ON` or `1` specifies the carrier frequency error is set automatically.

Examples `[SENSE]:AUDIO:CARRIER:OFFSET:AUTO ON` sets the carrier frequency error automatically.

[SENSe]:AUDio:DEMod:MUTE

Sets or queries the mute setting for the audio demodulation. Enabling Mute inhibits the audio from playing out through the PC speakers.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Sense group

Syntax
```
[SENSe]:AUDio:DEMod:MUTE { OFF | ON | 0 | 1 }
```

Arguments ON or 1 enables the Mute setting for audio demodulation.

OFF or 0 disables the Mute setting for audio demodulation.

Examples `[SENSE]:AUDIO:DEMOD:MUTE` 1 enables the audio demodulation mute setting, inhibiting the audio signal from playing out through the PC speakers.

[SENSe]:AUDio:DEMod[:STATe]

Sets or queries the audio demodulation state.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Sense group

Syntax `[SENSe]:AUDio:DEMod[:STATe] { 0 | 1 }`

Arguments 1 starts the demodulation.

0 stops the demodulation.

Examples `[SENSE]:AUDIO:DEMOD[:STATE]` 1 starts the demodulating the audio signal.

[SENSe]:AUDio:DEMod:STReam:ACTivate

Sets or queries the Stream to File function. When checked, anytime the Run button is pressed, the audio stream-to-file begins.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Sense group

Syntax
```
[SENSe]:AUDio:DEMod:STReam:ACTivate { OFF | ON | 0 | 1 }
[SENSe]:AUDio:DEMod:STReam:ACTivate?
```

Arguments ON or 1 enables the stream to file function.

OFF or 0 disables the stream to file function

Returns

Examples `[SENSE]:AUDIO:DEMOD:STREAM:ACTIVATE` 1 enables (activates) the audio streaming function.

`[SENSE]:AUDIO:DEMOD:STREAM:ACTIVATE`? might return 0, indicating the audio streaming function is deactivated.

[SENSe]:AUDio:DEMod:STReam[:FILE] (No Query Form)

Sets the file name and file location of the audio file when streaming to disc.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Sense group

Syntax `[SENSe]:AUDio:DEMod:STReam[:FILE] <file>`

Related Commands

Arguments <file_name>::=<string> specifies the file name and location. The file extension is .wav. You can omit the extension.

You can use the absolute path to specify the file name. For example, specify the SAMPLE1 file in the My Documents folder on the C drive as "C:\My Documents\Audio1".

If you omit the directory path, the default path is used, which is C:\SignalVu-PC Files\Sample Data Records initially.

Examples `[SENSE]:AUDIO:DEMOD:STREAM[:FILE]` C:\SignalVu-PC Files\Sample Data Records\Audio1 saves the streamed audio file Audio1 to the specified location.

[SENSe]:AUDio:DEMod:TYPE

Sets or queries both the audio signal demodulation type and bandwidth.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Sense group

Syntax `[SENSe]:AUDio:DEMod:TYPE { AM8K | FM8K | FM13K | FM75K | FM200K }`

Related Commands

Arguments AM8K

FM8K

FM13K

FM75K

FM200K

Returns Same as arguments.

Examples `[SENSE]:AUDIO:DEMOD:TYPE` AM8K sets the audio demodulation type to AM and the bandwidth to 8 KHz.

[SENSE]:AUDIO:DEMOD:TYPE? might return AM8K.

[SENSe]:AUDio:DEMod:VOLume

Sets or queries the volume of the demodulated audio signal when played out through the PC speakers.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Sense group

Syntax
```
[SENSe]:AUDio:DEMod:VOLume <value>
[SENSe]:AUDio:DEMod:VOLume?
```

Arguments <value>::=<NR2> the value of relative volume setting.

Range = 0.0 to 1.0.

Examples [SENSE]:AUDIO:DEMOD:VOLUME 1.0 sets the volume output to maximum.

[SENSe]:AUDio:FILTer:DEEMphasis

Sets or queries the de-emphasis filter type.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax
```
[SENSe]:AUDio:FILTer:DEEMphasis { U25 | U50 | U75 | U750 |
USER}
[SENSe]:AUDio:FILTer:DEEMphasis?
```

Arguments U25 specifies the 25 µs filter.

U50 specifies the 50 µs filter.

U75 specifies the 75 µs filter.

U750 specifies the 750 µs filter.

USER specifies a user-defined filter.

Examples `[SENSE]:AUDIO:FILTER:DEEMPHASIS U50` specifies the 50 µs filter for the audio measurement.

[SENSe]:AUDio:FILTer:DEEMphasis:ENABle

Enables or disables the de-emphasis filter for the audio measurement.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax
```
[SENSe]:AUDio:FILTer:DEEMphasis:ENABle { OFF | ON | 0 | 1 }
[SENSe]:AUDio:FILTer:DEEMphasis:ENABle?
```

Arguments `OFF` or `0` turns off the de-emphasis filter.

`ON` or `1` turns on the de-emphasis filter.

Examples `[SENSE]:AUDIO:FILTER:DEEMPHASIS:ENABLE ON` turns on the de-emphasis filter.

[SENSe]:AUDio:FILTer:DEEMphasis:USER

Sets or queries a user-defined de-emphasis audio filter.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax
```
[SENSe]:AUDio:FILTer:DEEMphasis:USER <value>
[SENSe]:AUDio:FILTer:DEEMphasis:USER?
```

Arguments <value> ::= <NR3> the value of the user defined filter.

Examples `[SENSE]:AUDIO:FILTER:DEEMPHASIS:USER` 50.0E-6 sets the user-defined de-emphasis filter to 50 µs.

[SENSe]:AUDio:FILTer:HPF

Sets or queries the high-pass filter type.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax

```
[SENSe]:AUDio:FILTer:HPF { H20 | H50 | H300 | H400 | USER }
[SENSe]:AUDio:FILTer:HPF?
```

Arguments

`H20` specifies the 20 Hz filter.

`H50` specifies the 50 Hz filter.

`H300` specifies the 300 Hz filter.

`H400` specifies the 400 Hz filter.

`USER` specifies a user-defined filter.

Examples `[SENSE]:AUDIO:FILTER:HPF H300` specifies the 300 Hz filter for the audio measurement.

[SENSe]:AUDio:FILTer:HPF:ENABle

Enables or disables the high-pass audio filter for the audio measurement.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax

```
[SENSe]:AUDio:FILTer:HPF:ENABle { OFF | ON | 0 | 1 }
[SENSe]:AUDio:FILTer:HPF:ENABle?
```

Arguments

`OFF` or `0` turns the high-pass audio filter off.

`ON` or `1` turns the high-pass audio filter on.

Examples `[SENSE]:AUDIO:FILTER:HPF:ENABLE ON` turns the high-pass audio filter on.

[SENSe]:AUDio:FILTer:HPF:USER

Sets or queries a user-defined high-pass audio filter.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax

```
[SENSe]:AUDio:FILTer:HPF:USER <value>
[SENSe]:AUDio:FILTer:HPF:USER?
```

Arguments <value> ::= <NR3> the value of the user defined filter.

Examples `[SENSE]:AUDIO:FILTER:HPF:USER 50` sets the user-defined high-pass filter to 50 Hz.

[SENSe]:AUDio:FILTer:LPF

Sets or queries the low-pass filter type.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax

```
[SENSe]:AUDio:FILTer:LPF { H300 | H3K | H15K | H30K | H80K
| H300K | USER }
[SENSe]:AUDio:FILTer:LPF?
```

Arguments

`H300` specifies the 300 Hz filter.

`H3K` specifies the 3 kHz filter.

`H15` specifies the 15 kHz filter.

`H30K` specifies the 30 kHz filter.

`H80K` specifies the 80 kHz filter.

`H300K` specifies the 300 kHz filter.

`USER` specifies a user-defined filter.

Examples `[SENSE]:AUDIO:FILTER:LPF H300` specifies the 300 Hz filter for the audio measurement.

[SENSe]:AUDio:FILTer:LPF:ENABle

Enables or disables the low-pass filter for the audio measurement.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax
```
[SENSe]:AUDio:FILTer:LPF:ENABle { OFF | ON | 0 | 1 }
[SENSe]:AUDio:FILTer:LPF:ENABle?
```

Arguments `OFF` or `0` turns the low-pass audio filter off.

`ON` or `1` turns the low-pass audio filter on.

Examples `[SENSE]:AUDIO:FILTER:LPF:ENABLE ON` turns the low-pass filter on.

[SENSe]:AUDio:FILTer:LPF:USER

Sets or queries a user-defined low-pass audio filter.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax
```
[SENSe]:AUDio:FILTer:LPF:USER <value>
[SENSe]:AUDio:FILTer:LPF:USER?
```

Arguments <value> ::= <NR3> the value of the user defined filter.

Examples `[SENSE]:AUDIO:FILTER:LPF:USER 1500` sets the user-defined low-pass filter to 1.50 kHz.

[SENSe]:AUDio:FILTer:MODE:PREDefined

Selects a predefined audio filter or a filter stored on the instrument file system.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax

```
[SENSe]:AUDio:FILTer:MODE:PREDefined { OFF | ON | 0 | 1 }
[SENSe]:AUDio:FILTer:MODE:PREDefined?
```

Arguments `OFF` or `0` specifies the predefined filter on the file system. The file is specified by the [SENSe]:AUDio:FILTer:SOURce? command.

`ON` or `1` specifies the filter will be one of the selected predefined filters.

Examples `[SENSE]:AUDIO:FILTER:MODE:PREDEFINED ON` specifies the filter will be one of the predefined filters.

[SENSe]:AUDio:FILTer:SOURce? (Query Only)

Queries the measurement filter from the user-defined source file.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax

```
[SENSe]:AUDio:FILTer:SOURce?
```

Related Commands [SENSe]:AUDio:FILTer:MODE:PREDefined

Arguments None

Returns <file name> ::= <string> the name of the predefined filter.

Examples `[SENSE]:AUDIO:FILTER:SOURCE? AUD-Test3` loads the predefined filter "AUD-Test3."

[SENSe]:AUDio:FILTer:STANdard

Sets or queries the Standard audio filter.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax
```
[SENSe]:AUDio:FILTer:STANdard { CCITT | CMSG }
[SENSe]:AUDio:FILTer:STANdard?
```

Arguments `CCITT` specifies the CCITT Standard filter.

`CMSG` specifies the CMSG Standard filter.

Examples `[SENSE]:AUDIO:FILTER:STANDARD CCITT` specifies the CCITT Standard filter.

[SENSe]:AUDio:FILTer:STANdard:ENABle

Enables or disables a Standard filter for the audio measurement.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax
```
[SENSe]:AUDio:FILTer:STANdard:ENABle { OFF | ON | 0 | 1 }
[SENSe]:AUDio:FILTer:STANdard:ENABle?
```

Arguments `OFF` or `0` turns the standard filter off.

`ON` or `1` turns the standard audio filter on.

Examples `[SENSE]:AUDIO:FILTER:STANDARD:ENABLE ON` turns the Standard filter on.

[SENSe]:AUDio:FREQuency

Sets or queries the reference audio frequency.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax
```
[SENSe]:AUDio:FREQuency <value>
[SENSe]:AUDio:FREQuency?
```

Arguments <value> ::= <NR3> the reference audio frequency.

Examples `[SENSE]:AUDIO:FREQUENCY?` might return `5.8240000000E+3` indicating that the reference audio frequency is 5.824 kHz.

[SENSe]:AUDio:FREQuency:AUTO

Determines whether to set the reference audio frequency automatically or manually.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax
```
[SENSe]:AUDio:FREQuency:AUTO { OFF | ON | 0 | 1 }
[SENSe]:AUDio:FREQuency:AUTO?
```

Arguments `OFF` or `0` specifies the audio reference frequency is set manually.

`ON` or `1` specifies the audio reference frequency is set automatically.

Examples `[SENSE]:AUDIO:FREQUENCY:AUTO ON` sets the audio reference frequency automatically.

[SENSe]:AUDio:HNOise:ENABle

Enables or disables the Hum and Noise information in the Audio Summary display.

Conditions Measurement views: Audio Summary

Group Sense commands

Syntax
```
[SENSe]:AUDio:HNOise:ENABle { OFF | ON | 0 | 1 }
[SENSe]:AUDio:HNOise:ENABle?
```

Related Commands [SENSe]:AUDio:HNOise:REFerence

Arguments `OFF` or `0` turns off the Hum and Noise information in Audio Summary display.

`On` or `1` turns on the Hum and Noise information in the Audio Summary display.

Examples `[SENSE]:AUDIO:HNOISE:ENABLE OFF` turns off the Hum and Noise information in the Audio Spectrum display.

[SENSe]:AUDio:HNOise:REFerence (No Query Form)

Captures the current Hum and Noise information in the Audio Summary display.

Conditions Measurement views: Audio Summary

Group Sense commands

Syntax
```
[SENSe]:AUDio:HNOise:REFerence
```

Related Commands [SENSe]:AUDio:HNOise:ENABle

Arguments None

Examples `[SENSE]:AUDIO:HNOISE:REFERENCE` shows the current Hum and Noise information in the Audio Summary display.

[SENSe]:AUDio:SIGnal:TYPE

Sets or queries the audio signal type.

The signal type selection determines the available selections for the [SENSe]:AUDio:UNITs command.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax
```
[SENSe]:AUDio:SIGnal:TYPE { AM | FM | DIRect | PM }
[SENSe]:AUDio:SIGnal:TYPE?
```

Related Commands [SENSe]:AUDio:UNITs

Arguments `AM` sets the signal type to amplitude modulation.

`FM` sets the signal type to frequency modulation.

`DIRect` sets the signal type to direct modulation.

`PM` sets the signal type to pulse modulation.

Returns

Examples `[SENSE]:AUDIO:SIGNAL:TYPE FM` sets the signal type to FM, frequency modulation for the audio spectrum measurement.

[SENSe]:AUDio:SPECtrum:CLEar:RESults (No Query Form)

Clears the results of the audio spectrum measurement.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax
```
[SENSe]:AUDio:SPECtrum:CLEar:RESults
```

Arguments None

[SENSe]:AUDio:SPECtrum:POINts:COUNt

Sets or queries the number of trace points acquired for the audio spectrum display.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax
```
[SENSe]:AUDio:SPECtrum:POINts:COUNt { P801 | P2401 | P4001 | P8001 | P10401 }
[SENSe]:AUDio:SPECtrum:POINts:COUNt?
```

Arguments `P801` sets the number of sample points to 801.

`P2401` sets the number of sample points to 2401.

`P4001` sets the number of sample points to 4001.

`P8001` sets the number of sample points to 8001.

`P10401` sets the number of sample points to 10401.

Examples `[SENSE]:AUDIO:SPECTRUM:POINTS:COUNT P2401` sets the number of sample points to 2401 in the audio spectrum display.

[SENSe]:AUDio:UNITs

Sets or queries the units of the audio spectrum measurement.

The available selections depend on signal type.

Conditions Measurement views: Audio Spectrum

Group Sense commands

Syntax
```
[SENSe]:AUDio:UNITs { AM | AM2 | DBAM2 | HZ | HZ2 | DBHZ2 | RAD | RAD2 | DBRAD2 | DBM | DBV | VOLTs | WATTs }
[SENSe]:AUDio:UNITs?
```

Related Commands [SENSe]:AUDio:SIGnal:TYPE

Arguments The following table lists the arguments.

Argument	Unit	Signal type
DBM	dBm	Direct
DBV	dBV	
VOLTs	Volts	
WATTs	Watts	

Argument	Unit	Signal type
AM	am	AM
AM2	am^2	
DBAM2	$dBam^2$	
HZ	Hz	FM
HZ2	Hz^{22}	
DBHZ2	$dBHz^2$	
RAD	rad	PM
RAD2	rad^2	
DBRAD2	$dBrad^2$	

Examples `[SENSE]:AUDIO:UNITS VOLTs` sets the units to volts for the Direct signal type.

[SENSe]:AVTime:{BANDwidth|BWIDth}

Sets or queries the time-domain bandwidth filter in the Amplitude versus Time measurement. Programming a specified bandwidth disables the [SENSe]:AVTime:SPAN setting.

Conditions Measurement views: Amplitude versus Time

Group Sense commands

Syntax
```
[SENSe]:AVTime:{BANDwidth|BWIDth} <value>
[SENSe]:AVTime:{BANDwidth|BWIDth}?
```

Arguments `<value> ::= <NRf>` specifies the filter bandwidth.
Range: 1 Hz to 20 MHz (Standard) / 60 MHz (Option 110).

Examples `SENSE:AVTIME:BANDWITH 10MHz` sets the filter bandwidth to 10 MHz.

[SENSe]:AVTime:{BANDwidth|BWIDth}:ACTual? (Query Only)

Queries the actual time-domain bandwidth in the Amplitude versus Time measurement.

Conditions Measurement views: Amplitude versus Time

Group Sense commands

Syntax `[SENSe]:AVTime:{BANDwidth|BWIDth}:ACTual?`

Arguments None

Returns `<NRf>` The actual time-domain bandwidth in Hz.

Examples `SENSE:AVTIME:BANDWIDTH:ACTUAL?` might return `20E+6`, indicating that the actual time-domain bandwidth is 20 MHz.

[SENSe]:AVTime:CLEar:RESults (No Query Form)

Restarts multi-trace functions (Average and Max/Min Hold).

Conditions Measurement views: Amplitude versus Time

Group Sense commands

Syntax `[SENSe]:AVTime:CLEar:RESults`

Arguments None

Examples `SENSE:AVTIME:CLEAR:RESULTS` restarts multi-trace functions.

[SENSe]:AVTime:MAXTracepoints

Sets or queries the maximum trace points in the Amplitude versus Time measurement.

Conditions Measurement views: Amplitude versus Time

Group Sense commands

Syntax `[SENSe]:AVTime:MAXTracepoints { ONEK | TENK | HUNDredk | NEVerdecimate }`

```
[SENSe]:AVTime:MAXTracepoints?
```

Arguments `ONEK` sets the maximum trace points to 1 k.

`TENK` sets the maximum trace points to 10 k.

`HUNDredk` sets the maximum trace points to 100 k.

`NEVerdecimate` never decimates the trace points.

Examples `SENSE:AVTIME:MAXTRACEPOINTS TENK` sets the maximum trace points to 10 k.

[SENSe]:AVTime:METHod

Sets or queries the method to set the measurement bandwidth in the Amplitude versus Time measurement.

Conditions Measurement views: Amplitude versus Time

Group Sense commands

Syntax

```
[SENSe]:AVTime:METHod { SPAN | TDBW }
[SENSe]:AVTime:METHod?
```

Arguments `SPAN` specifies that the measurement bandwidth is set by the frequency span, using the [SENSe]:AVTime:SPAN command.

`TDBW` specifies that the measurement bandwidth is set by the time-domain bandwidth, using the [SENSe]:AVTime:{BANDwidth|BWIDth} command.

Examples `SENSE:AVTIME:METHOD SPAN` specifies that the measurement bandwidth is set by the frequency span.

[SENSe]:AVTime:SPAN

Sets or queries the frequency span in the Amplitude versus Time measurement. Programming a specified span disables the [SENSe]:AVTime:{BANDwidth|BWIDth} setting.

Conditions Measurement views: Amplitude versus Time

Group Sense commands

Syntax
```
[SENSe]:AVTime:SPAN <value>
[SENSe]:AVTime:SPAN?
```

Arguments <value> :: <NRf> specifies the frequency span.
Range: 10 Hz to 40 MHz (Standard) / 110 MHz (Option 110)

Examples SENSE:AVTIME:SPAN 5MHz sets the frequency span to 5 MHz.

[SENSe]:BLUEtooth:ANALysis:LENGth

Sets or returns the analysis length in the Bluetooth displays. Programming a specified length sets [SENSe]:BLUEtooth:ANALysis:LENGth:AUTO to off.

Conditions Measurement views: All Bluetooth displays

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:ANALysis:LENGth <value>
[SENSe]:BLUEtooth:ANALysis:LENGth?
```

Related Commands [SENSe]:BLUEtooth:ANALysis:LENGth:AUTO

Arguments <value> ::= <NRf> the analysis length. The minimum range depends on the modulation type.

Examples [SENSE]:BLUETOOTH:ANALYSIS:LENGTH 25.625E-6 sets the analysis length to 25.625 µs.

[SENSe]:BLUEtooth:ANALysis:LENGth:ACTual? (Query Only)

Sets the actual analysis length in the Bluetooth displays.

Conditions Measurement views: All Bluetooth displays

Group Sense commands

Syntax `[SENSe]:BLUEtooth:ANALysis:LENGth:ACTual?`

Arguments None

Returns <NR3> the actual analysis length in seconds.

Examples `[SENSE]:BLUETOOTH:ANALYSIS:LENGTH:ACTUAL?` might return 25.625E-6, indicating that the actual analysis length is 25.625 µs.

[SENSe]:BLUEtooth:ANALysis:LENGth:AUTO

Sets or queries the length mode of the Bluetooth displays.

Conditions Measurement views: All Bluetooth displays

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:ANALysis:LENGth:AUTO { OFF | ON | 0 | 1 }
[SENSe]:BLUEtooth:ANALysis:LENGth:AUTO?
```

Related Commands [SENSe]:BLUEtooth:ANALysis:LENGth

Arguments OFF or 0 sets the analysis length manually, using the [SENSe]:BLUEtooth:ANALysis:LENGth command.

ON or 1 sets the analysis length automatically.

Examples `[SENSE]:BLUETOOTH:ANALYSIS:LENGTH:AUTO` ON sets the analysis length mode automatically.

[SENSe]:BLUEtooth:ANALysis:MEAS[:BANDwidth|BWIDth]

Sets or returns the measurement bandwidth in the Bluetooth measurement.

Conditions Measurement views: All Bluetooth displays

Group Sense commands

Syntax

```
[SENSe]:BLUEtooth:ANALysis:MEAS[:BANDwidth|BWIDth] <value>
[SENSe]:BLUEtooth:ANALysis:MEAS[:BANDwidth|BWIDth]?
```

Arguments

<value> ::= <NR3>, the measurement bandwidth.

Examples

`[SENSE]:BLUETOOTH:ANALYSIS:MEAS[:BANDWIDTH|BWIDTH]`? might return 6.400E+6 indicating the measurement bandwidth is 6.400 MHz.

[SENSe]:BLUEtooth:ANALysis:MEAS:SOURce

Sets or returns the analysis measurement source in the Bluetooth measurement.

Conditions

Measurement views: All Bluetooth displays

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth:ANALysis:MEAS:SOURce { MANual | AUTO | LSP
}
[SENSe]:BLUEtooth:ANALysis:MEAS:SOURce?
```

Arguments

MANual allows for manual selection of the measurement bandwidth. A typical use is setting a narrow measurement bandwidth on a small range of frequencies to examine one signal while rejecting the others that are present in a Spectrum graph.

AUTO automatically selects a measurement bandwidth. The measurement picks a bandwidth based on other parameter settings (such as symbol rate and modulation type).

LSP sets the measurement bandwidth to Link to Span. This allows you to use the Spectrum display to tune the frequency, view the signal, and then use the Span control to set the measurement bandwidth. This mode emulates legacy instruments.

Examples

`[SENSE]:BLUETOOTH:ANALYSIS:MEAS:SOURCE`? might return MAN indicating a manual selection of the bandwidth measurement.

[SENSe]:BLUEtooth:ANALysis:OFFSet

Sets or returns the Bluetooth symbol analysis offset.

Conditions

Measurement views: Bluetooth displays.

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:ANALysis:OFFSet <value>
[SENSe]:BLUEtooth:ANALysis:OFFSet?
```

Arguments <NR1> the symbol analysis offset as a percentage. The range is from 0 to 100%.

Examples `[SENSE]:BLUETOOTH:ANALYSIS:OFFSET` -50 sets the symbol analysis of set to -50%.

[SENSe]:BLUEtooth:ANALysis:OFFSet:AUTO

Determines whether to set the carrier frequency error automatically or manually in the Bluetooth displays.

Conditions Measurement views: Bluetooth displays.

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:ANALysis:OFFSet:AUTO { OFF | ON | 0 | 1 }
[SENSe]:BLUEtooth:ANALysis:OFFSet:AUTO?
```

Arguments OFF or 0 specifies the carrier frequency error is set manually.
ON or 1 specifies the carrier frequency error is set automatically.

Examples `[SENSE]:BLUETOOTH:ANALYSIS:OFFSET:AUTO` ON sets the carrier frequency error automatically.

[SENSe]:BLUEtooth:ANALysis:TIMe:UNITs (No Query Form)

Sets the time units for the Bluetooth Constellation settings.

Conditions delete if no conditions

Group Sense commands

Syntax

```
[SENSe]:BLUEtooth:ANALysis:TIMe:UNITs {SEConds | SYMBols }
[SENSe]:BLUEtooth:ANALysis:TIMe:UNITs?
```

Arguments

SEConds sets the time units to seconds.

SYMBols sets the time units to symbols.

Examples

`[SENSE]:BLUETOOTH:ANALYSIS:TIME:UNITS` sets the time units to seconds.

[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):MODE

Sets or returns the measurement bandwidth mode for the Bluetooth measurements.

Conditions

Measurement views: Any Bluetooth display

Group

Sense commands

Syntax

```
[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):MODE { MANual | AUTO
|LSP }
[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):MODE?
```

Arguments

MANual allows for manual selection of the measurement bandwidth. A typical use is setting a narrow measurement bandwidth on a small range of frequencies to examine one signal while rejecting the others that are present in a Spectrum graph.

AUTO automatically selects a measurement bandwidth. The measurement picks a bandwidth based on other parameter settings (such as symbol rate and modulation type).

LSP sets the Measurement BW to Link to Span. This allows you to use the Spectrum display to tune the frequency, view the signal, and then use the Span control to set the measurement bandwidth. This mode emulates legacy instruments.

Examples

`[SENSE]:BLUETOOTH([:BANDWIDTH]|:BWIDTH):MODE` AUTO sets the bandwidth mode to automatically select a measurement bandwidth.

[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):TINTerval

Sets or returns the measurement bandwidth frequency span in the Bluetooth displays.

Conditions Measurement views: Any Bluetooth display

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):TINTerval <value>
[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):TINTerval?
```

Arguments <value> ::= <NRf> specifies the measurement bandwidth in MHz.

Returns <NRf> the actual measurement bandwidth.

Examples `[SENSE]:BLUETOOTH([:BANDWIDTH]|:BWIDTH):TINTERVAL?` might return 35.255E+6 indicating the bandwidth is 35.255 MHz.

[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):TINTerval:AUTO

Sets or returns the Bluetooth measurement bandwidth to Auto or to Manual.

Conditions Measurement views: Any Bluetooth display

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):TINTerval:AUTO { 0
| 1 | OFF | ON }
[SENSe]:BLUEtooth([:BANDwidth]|:BWIDth):TINTerval:AUTO?
```

Arguments OFF or 0 sets measurement bandwidth to Auto.

ON or 1 sets measurement bandwidth to Manual.

Examples `[SENSE]:BLUETOOTH([:BANDWIDTH]|:BWIDTH):TINTERVAL:AUTO?` might return 1 indicating the measurement bandwidth is set to Auto.

[SENSe]:BLUEtooth:CONStellation:PREFs:GRATicule:SHOW

Show or hides the graticule state in the Bluetooth Constellation display.

Conditions Measurement views: Bluetooth Constellation

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:CONStellation:PREFs:GRATicule:SHOW { OFF | ON | 0 | 1 }
[SENSe]:BLUEtooth:CONStellation:PREFs:GRATicule:SHOW?
```

Arguments OFF or 0 hides the graticule in the display.

ON or 1 shows the graticule in the display.

Returns 0 indicates the graticule is turned off.

1 indicates the graticule is turned on.

Examples `[SENSE]:BLUETOOTH:CONSTELLATION:PREFS:GRATICULE:SHOW` ON turns on the graticule in the display.

[SENSe]:BLUEtooth:CONStellation:PREFs:MARKERS:SHOW

Show or hides the marker readout in the Bluetooth Constellation display.

Conditions Measurement views: Bluetooth Constellation

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:CONStellation:PREFs:MARKERS:SHOW { OFF | ON | 0 | 1 }
[SENSe]:BLUEtooth:CONStellation:PREFs:MARKERS:SHOW?
```

Arguments OFF or 0 hides the marker readout in the display.

ON or 1 shows the marker readout in the display.

Returns 0 indicates the marker readout is turned off.

1 indicates the marker readout is turned on.

Examples `[SENSE]:BLUETOOTH:CONSTELLATION:PREFS:MARKERS:SHOW` ON turns on the marker readout in the display.

[SENSe]:BLUEtooth:CONStellation:PREFs:RADIx

Sets or returns the radix of the marker readout in the Bluetooth Constellation display.

Conditions Measurement views: Bluetooth Constellation

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:CONStellation:PREFs:RADIx { Binary | Hex }
[SENSe]:BLUEtooth:CONStellation:PREFs:RADIx?
```

Arguments Binary sets the marker readout to Binary.

Hex sets the marker readout to Hex.

Examples `[SENSE]:BLUETOOTH:CONSTELLATION:PREFS:RADIX?` might return BINARY indicating the Marker readout radix is set to Binary.

[SENSe]:BLUEtooth:CONStellation:TRACE<x>:CONTent

Sets or returns the appearance of the specified trace (Trace 1 or Trace 2) in the Bluetooth Constellation display.

Conditions Measurement views: Bluetooth Constellation

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:CONStellation:TRACE<x>:CONTent { Vectors |
Points | Lines }
[SENSe]:BLUEtooth:CONStellation:TRACE<x>:CONTent?
```

Arguments Vectors indicates the traces appear as vectors.

Points indicates the traces appear as a series of points.

Lines indicates the traces appear as lines.

Examples `SENSE:BLUETOOTH:CONSTELLATION:TRACE1:CONTENT VECTORS` sets the Trace 1 to show as vectors in the display.

[SENSe]:BLUEtooth:CONStellation:TRACe<x>:FREEze

Sets or freezes the updates of the specified trace (Trace 1 or Trace 2) in the Bluetooth Constellation display.

Conditions Measurement views: Bluetooth Constellation

Group Sense commands

Syntax

```
[SENSe]:BLUEtooth:CONStellation:TRACe<x>:FREEze { OFF | ON
| 0 | 1 }
[SENSe]:BLUEtooth:CONStellation:TRACe<x>:FREEze?
```

Arguments OFF or 0 normally updates the display of the specified trace.

ON or 1 stops updating the display of the specified trace.

Examples `SENSE:BLUETOOTH:CONSTELLATION:TRACE1:FREEZE1` freezes the updates of Trace 1.

[SENSe]:BLUEtooth:CONSTellation:TRACE<x>:POINtsPerSymbol

Sets or returns the number of points per symbol (how many points to use between symbols) for the specified trace (Trace 1 or Trace 2) in the Bluetooth Constellation display.

Conditions Measurement views: Bluetooth Constellation

Group Sense commands

Syntax

```
[SENSe]:BLUEtooth:CONSTellation:TRACE<x>:POINtsPerSymbol {
ONE |TWO | FOUR | EIGHt | SIXTeen | THIRtytwo }
[SENSe]:BLUEtooth:CONSTellation:TRACE<x>:POINtsPerSymbol?
```

Arguments The number of points per symbol (1,2,4,8,16,32).

Examples `SENSE:BLUETOOTH:CONSTELLATION:TRACE2:POINTSPERSYMBOL FOUR` sets the number of points per symbol for trace 2 to four.

[SENSe]:BLUEtooth:CONStellation:TRACe<x>:SHOW

Shows or hides the specified trace (Trace 1 or Trace 2) in the Bluetooth Constellation display.

Conditions Measurement views: Bluetooth Constellation

Group Sense commands

Syntax

```
[SENSe]:BLUEtooth:CONStellation:TRACe<x>:SHOW { OFF | ON |
0 | 1 }
[SENSe]:BLUEtooth:CONStellation:TRACe<x>:SHOW?
```

Arguments OFF or 0 hides the specified trace in the display.

ON or 1 shows the specified trace in the display.

Examples `SENSE:BLUETOOTH:CONSTELLATION:TRACE1:SHOW OFF` hides trace 1 in the display.

[SENSe]:BLUEtooth:DELTa:AVERage:FONE

Sets or returns the average frequency for the low deviation pattern in the Bluetooth measurement.

Conditions Measurement views: Any Bluetooth display

Group Sense commands

Syntax

```
[SENSe]:BLUEtooth:DELTa:AVERage:FONE <value>
[SENSe]:BLUEtooth:DELTa:AVERage:FONE?
```

Arguments <value> ::= <NR3>, the average frequency of the low deviation pattern.

Examples `[SENSE]:BLUETOOTH:DELTA:AVERAGE:FONE?` might return 139.8309531250E+3 indicating the average frequency of the low deviation pattern is 139.83 kHz.

[SENSe]:BLUEtooth:DELTa:AVERage:FTWO

Sets or returns the average frequency for the high deviation pattern in the Bluetooth measurement.

Conditions Measurement views: Any Bluetooth display

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:DELTa:AVERage:FTWO <value>
[SENSe]:BLUEtooth:DELTa:AVERage:FTWO?
```

Arguments <value> ::= <NR3>, the average frequency of the high deviation pattern.

Examples `[SENSE]:BLUETOOTH:DELTA:AVERAGE:FTWO?` might return 139.8309531250E+3 indicating the average frequency of the low deviation pattern is 139.83 kHz.

[SENSe]:BLUEtooth:FDVTime:TRACE:OCTEt? (Query Only)

Returns the selected octet number on the Frequency Deviation vs. Time display.

Conditions Measurement views: Frequency Deviation vs. Time

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:FDVTime:TRACE:OCTEt?
```

Arguments None

Returns <NR3> the octet number.

Examples `SENSe:BLUEtooth:FDVTime:TRACE:OCTEt?` might return 5 indicating that 5 is the selected octet number.

[SENSe]:BLUEtooth:FDVTime:TRACE:OCTEt:TOTAL:COUNt? (Query Only)

Returns the total octet value to be selected on the Frequency Deviation vs. Time display.

For the standard low-energy range the maximum number of octets available is 36. For standard basic rate range, the maximum number of octets depends on the packet type.

Conditions Measurement views: Frequency Deviation vs. Time

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:FDVTime:TRACE:OCTEt:TOTAL:COUNt?
```

Arguments None

Returns An <NR3> number indicating the maximum numbers of octets available.

Examples `SENSe:BLUEtooth:FDVTime:TRACE:OCTEt:TOTAL:COUNt?` might return 36 indicating the maximum number of octets selected is 36.

[SENSe]:BLUEtooth:FDVTime:TRACE:VIEW:MODE

Sets or queries view mode in the Frequency Deviation vs. Time display.

Conditions Measurement views: Frequency Deviation vs. Time

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:FDVTime:TRACE:VIEW:MODE {PACK | OCTET}
[SENSe]:BLUEtooth:FDVTime:TRACE:VIEW:MODE?
```

Arguments PACK indicates the display is in Packet mode.

OCTET indicates the display is in Octet mode.

Examples `SENSe:BLUEtooth:FDVTime:TRACE:VIEW:MODE PACK` sets the view to packet mode.

[SENSe]:BLUEtooth:FILTer:ALPHa

Sets or returns the alpha filter factor (α/BT) in the Bluetooth analysis.

Conditions Measurement views: Any Bluetooth display

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:FILTer:ALPHa <value>
[SENSe]:BLUEtooth:FILTer:ALPHa?
```

Arguments <value> ::= <NRf> the filter factor; range: 0.001 to 1.

Examples `[SENSE]:BLUETOOTH:FILTER:ALPHA` 0.5 sets the filter factor to 0.5.

[SENSe]:BLUEtooth:FILTer:MEASurement

Sets or returns the measurement filter in the Bluetooth measurements.

Conditions Measurement views: Any Bluetooth display

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:FILTer:MEASurement {OFF | BR | LE | USER1
| USER2| UOTH }
[SENSe]:BLUEtooth:FILTer:MEASurement?
```

Arguments
OFF no measurement filter is used.
BR, the Basic Rate filter is used.
LE, the Low Energy filter is used.
USER1, the User Meas Filter 1 is used.
USER2, the User Meas Filter 2 is used.
UOTH a user-defined filter is used.

Examples `[SENSE]:BLUETOOTH:FILTER:MEASUREMENT` USER1 uses the User Measurement Filter 1 as the measurement filter.

[SENSe]:BLUEtooth:FILTer:REFerence

Sets or returns the reference filter used in the Bluetooth measurements.

Conditions Measurement views: Any Bluetooth display

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:FILTer:REFerence {OFF | BR | USER1 |
USER2| UOTH }
[SENSe]:BLUEtooth:FILTer:REFerence?
```

Related Commands AADD

Arguments OFF no reference measurement filter is used.
BR, the Basic Rate filter is used.
USER1, the User Ref Filter 1 is used.
USER2, the User Ref Filter 2 is used.
UOTH, a user-defined filter is used.

Examples `[SENSE]:BLUETOOTH:FILTER:REFERENCE` OFF selects None or no reference filter.

[SENSe]:BLUEtooth:FREQuency:DEViation

Sets or returns the frequency deviation in the Bluetooth analysis.

Conditions Measurement views: Any Bluetooth display

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:FREQuency:DEViation <value>
[SENSe]:BLUEtooth:FREQuency:DEViation?
```

Related Commands [SENSe]:BLUEtooth:FREQuency:DEViation:AUTO

Arguments <value> ::= <NRf>, the frequency deviation.

The Standard product range is 1 Hz to 40 MHz; the Option 110 product range is 1 Hz to 110 MHz.

Examples `[SENSE]:BLUETOOTH:FREQUENCY:DEVIATION` 1MHz sets the frequency deviation to 1 MHz.

[SENSe]:BLUEtooth:FREQuency:DEViation:AUTO

Determines whether to detect the frequency deviation automatically or manually for the Bluetooth measurement.

Conditions Measurement views: Any Bluetooth display

Group Sense commands

Syntax

```
[SENSe]:BLUEtooth:FREQuency:DEViation:AUTO { 0 | 1 | OFF | ON }
[SENSe]:BLUEtooth:FREQuency:DEViation:AUTO?
```

Related Commands [SENSe]:BLUEtooth:FREQuency:DEViation

Arguments OFF or 0 sets the frequency deviation manually.

ON or 1 automatically calculates the frequency deviation; (default setting).

Examples `[SENSE]:BLUETOOTH:FREQUENCY:DEVIATION:AUTO` ON automatically calculates the frequency deviation.

[SENSe]:BLUEtooth:FREQuency:ERRor

Sets or returns the frequency error in the Bluetooth analysis parameters (when Auto is off).

Conditions Measurement views: Any Bluetooth display

Group Sense commands

Syntax

```
[SENSe]:BLUEtooth:FREQuency:ERRor <value>
[SENSe]:BLUEtooth:FREQuency:ERRor?
```

Related Commands [SENSe]:BLUEtooth:FREQuency:ERRor:AUTO

Arguments <value> ::= <NRf> the frequency error in Hz.

Examples `[SENSE]:BLUETOOTH:FREQUENCY:ERROR` 0.02 sets the frequency error to 0.02.

[SENSe]:BLUEtooth:FREQuency:ERRor:AUTO

Determines whether the frequency error is set to Auto or Manual.

To enter a manual value, when Auto is Off, use the [SENSe]:BLUEtooth:FREQuency:ERRor command.

Conditions Measurement views: Any Bluetooth display

Group Sense commands

Syntax

```
[SENSe]:BLUEtooth:FREQuency:ERRor:AUTO { 0 | 1 | OFF | ON }
[SENSe]:BLUEtooth:FREQuency:ERRor:AUTO?
```

Arguments OFF or 0 sets the frequency error manually using the [SENSe]:BLUEtooth:FREQuency:ERRor command.

ON or 1 sets the frequency error automatically.

Examples `[SENSE]:BLUETOOTH:FREQUENCY:ERROR:AUTO` ON determines the frequency error automatically.

[SENSe]:BLUEtooth:INBEmissions:POWEr:LIMIt<x>

Sets or queries the limits of the Bluetooth InBand Emission settings.

When <x> = 1, then commands set or query the fTX ± 2 MHz limit.

When <x> = 2, then commands set or query the fTX ± (3 + n) MHz limit.

Conditions Measurement views: Bluetooth InBand Emissions display

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:INBEmissions:POWEr:LIMIt<x> <value>
[SENSe]:BLUEtooth:INBEmissions:POWEr:LIMIt<x>
```

Arguments <value> ::= <NR3> the limit of the specified inband emissions setting in dBm.

Examples SENSE:BLUETOOTH:INBEMISSIONS:POWER:LIMIT2? might return -20 indicating the fTX ± 2 limit is -20 dBm.

[SENSe]:BLUEtooth:POWer:CLASs

Sets or queries the power class of the Bluetooth standard.

Conditions Measurement views: Bluetooth displays

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:POWer:CLASs {THREE|TWO|ONE}
[SENSe]:BLUEtooth:POWer:CLASs?
```

Arguments THREE is power class 3.
TWO is power class 2.
ONE is power class 1.

Examples `[SENSE]:BLUETOOTH:POWER:CLASS` THREE sets the power class to 3.

[SENSe]:BLUEtooth:STANdard

Sets or queries the Bluetooth standard.

Conditions Measurement views: Bluetooth Constellation

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:STANdard {BR|LE|EDR}
[SENSe]:BLUEtooth:STANdard?
```

Arguments BR is the Basic Rate standard.
LE is the Low Energy standard.
EDR is the Eye Diagram standard.

Examples `[SENSE]:BLUETOOTH:STANDARD` BR sets the standard type to Basic Rate.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FONE:AVERage:MAX

Sets or returns the maximum limit of the Basic Rate ΔF1 average in the Bluetooth Constellation Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FONE:AVERage:MAX
<value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FONE:AVERage:MAX?
```

Arguments <value> ::= <NRf> the maximum average limit in Hz.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BDR:DELTA:FONE:AVERAGE:MAX` 175.0E+3 sets the maximum Basic Rate limit of the ΔF1 average to 175.0 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FONE:AVERage:MIN

Sets or returns the minimum limit of the Basic Rate ΔF1 average in the Bluetooth Constellation Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FONE:AVERage:MIN
<value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FONE:AVERage:MIN?
```

Arguments <value> ::= <NRf> the minimum average limit in Hz.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BDR:DELTA:FONE:AVERAGE:MIN` 140.0E+3 sets the minimum limit of the Basic Rate ΔF1 average to 140.0 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FTWO:PERcent

Sets or returns the limit of the Basic Rate ΔF2 percentage in the Bluetooth Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FTWO:PERcent
<value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FTWO:PERcent?
```

Arguments <value> ::= <NRf> the limit in percent (%).

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BDR:DELTA:FTWO:PERCENT`? might return 99.00 indicating the limit of the Basic Rate ΔF2 percentage is 99%.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FTWO:RESult

Sets or returns the limit of the Basic Rate ΔF2 result in the Bluetooth Constellation Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FTWO:RESult
<value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:DELTa:FTWO:RESult?
```

Arguments <value> ::= <NRf> the limit in Hz.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BDR:DELTA:FTWO:RESULT` 115.0E+3 sets the limit of the Basic Rate ΔF2 result to 115.0 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:DRIFt:FNFZero

Sets or returns the limit of the Basic Rate maximum drift fn–f0 in the Bluetooth Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:DRIFt:FNFZero
<value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:DRIFt:FNFZero?
```

Arguments <value> ::= <NRf> the limit in Hz.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BDR:MAX:DRIFT:FNFZERO?` might return 20.00E+3 indicating the limit of the Basic Rate maximum drift fn–f0 is 20 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:FREQuency:OFFSet

Sets or returns the limit of the Basic Rate maximum frequency offset in the Bluetooth Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:FREQuency:OFFSet
<value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:FREQuency:OFFSet?
```

Arguments <value> ::= <NRf> the limit in Hz.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BDR:MAX:FREQUENCY:OFFSET` 150.0E+3 sets the Basic Rate maximum frequency offset limit to 150 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:FREQuency:OFFSet:PREAmBDR

Sets or returns the limit of the Basic Rate maximum frequency offset value from the preamble region in the Bluetooth Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:FREQuency:OFFSet:
PREAmBDR <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:MAX:FREQuency:OFFSet:
PREAmBDR?
```

Arguments <value> ::= <NRf> the limit in Hz.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BDR:MAX:FREQUENCY:OFFSET:PREAMBDR` 150.0E+3 sets the limit of the Basic Rate maximum frequency offset limit to 150 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:AVERage:MAX

Sets or returns the limit of the Basic Rate Power Class 1 maximum average output power value in the Bluetooth Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:
AVERage:MAX <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:
AVERage:MAX?
```

Arguments <value> ::= <NRf> the limit in dBm.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:ONE:AVERAGE:MAX?` might return 20.00 indicating the maximum output power average is 20.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:AVERage:MIN

Sets or returns the limit of the Basic Rate Power Class 1 minimum average output power value in the Bluetooth Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:
AVERage:MIN <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:
AVERage:MIN?
```

Arguments <value> ::= <NRf> the limit in dBm.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:ONE:AVERAGE:MIN?` might return 0.00 indicating the minimum output power average is 0 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:PEAK:MAX

Sets or returns the limit of the Basic Rate Power Class 1 maximum peak output power value in the Bluetooth Constellation Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:PEAK:
MAX <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:PEAK:
MAX?
```

Arguments <value> ::= <NRf> the limit in dBm.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:ONE:PEAK:MAX?` might return 23.00 indicating the maximum peak output power is 23.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:PEAK:MIN

Sets or returns the limit of the Basic Rate Power Class 1 minimum peak output power value in the Bluetooth Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:PEAK:MIN <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:ONE:PEAK:MIN?
```

Arguments <value> ::= <NRf> the limit in dBm.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:ONE:PEAK:MIN?` might return 3.00 indicating the maximum peak output power is 3.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:AVERage:MAX

Sets or returns the limit of the Basic Rate Power Class 3 maximum average output power value in the Bluetooth Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:AVERage:MAX <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:AVERage:MAX?
```

Arguments <value> ::= <NRf> the limit in dBm.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:THREE:AVERAGE:MAX?` might return -40.00 indicating the maximum output power average is -40.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:AVERage:MIN

Sets or returns the limit of the Basic Rate Power Class 3 minimum average output power value in the Bluetooth Constellation Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:
AVERage:MIN <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:
AVERage:MIN?
```

Arguments <value> ::= <NRf> the limit in dBm.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:THREE:AVERAGE:MIN?` might return 0.00 indicating the minimum output power average is 0.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:PEAK:MAX

Sets or returns the limit of the Basic Rate Power Class 3 maximum peak output power value in the Bluetooth Constellation Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:PEAK:
MAX <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:PEAK:
MAX?
```

Arguments <value> ::= <NRf> the limit in dBm.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:THREE:PEAK:MAX?` might return 37.00 indicating the maximum peak output power is 37.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:PEAK:MIN

Sets or returns the limit of the Basic Rate Power Class 3 minimum peak output power value in the Bluetooth Constellation Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:PEAK:MIN <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:THREe:PEAK:MIN?
```

Arguments <value> ::= <NRf> the limit in dBm.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:THREE:PEAK:MIN?` might return 3.00 indicating the maximum peak output power is 3.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:AVERage:MAX

Sets or returns the limit of the Basic Rate Power Class 2 maximum average output power value in the Bluetooth Constellation Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:AVERage:MAX <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:AVERage:MAX?
```

Arguments <value> ::= <NRf> the limit in dBm.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:TWO:AVERAGE:MAX?` might return 4.00 indicating the maximum output power average is 4.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:AVERage:MIN

Sets or returns the limit of the Basic Rate Power Class 2 minimum average output power value in the Bluetooth Constellation Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:
AVERage:MIN <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:
AVERage:MIN?
```

Arguments <value> ::= <NRf> the limit in dBm.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:TWO:AVERAGE:MIN?` might return -6.00 indicating the minimum output power average is -6.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:PEAK:MAX

Sets or returns the limit of the Basic Rate Power Class 2 maximum peak output power value in the Bluetooth Constellation Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:PEAK:
MAX <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:PEAK:
MAX?
```

Arguments <value> ::= <NRf> the limit in dBm.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:TWO:PEAK:MAX?` might return 7.00 indicating the maximum peak output power is 7.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:PEAK:MIN

Sets or returns the limit of the Basic Rate Power Class 2 minimum peak output power value in the Bluetooth Constellation Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:PEAK:MIN <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BDR:OPOWer:CLASs:TWO:PEAK:MIN?
```

Arguments <value> ::= <NRf> the limit in dBm.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BDR:OPOWER:CLASS:TWO:PEAK:MIN?` might return -3.00 indicating the maximum peak output power is -3.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FONE:AVERage:MAX

Sets or returns the maximum limit of the Low Energy ΔF1 average in the Bluetooth Constellation Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FONE:AVERage:MAX <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FONE:AVERage:MAX?
```

Arguments <value> ::= <NRf> the maximum average limit in Hz.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BLE:DELTA:FONE:AVERAGE:MAX` 275.0E+3 sets the maximum Low Energy limit of the Low Energy ΔF1 average to 275.0 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FONE:AVERage:MIN

Sets or returns the minimum limit of the Low Energy ΔF1 average in the Bluetooth Constellation Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FONE:AVERage:MIN
<value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FONE:AVERage:MIN?
```

Arguments <value> ::= <NRf> the minimum average limit in Hz.

Examples
```
[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BLE:DELTA:FONE:AVERAGE:MIN
```
225.0E+3 sets the minimum limit of the Low Energy ΔF1 average to 225.0 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FTWO:PERcent

Sets or returns the limit of the Low Energy ΔF2 percentage in the Bluetooth Constellation Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FTWO:PERcent
<value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FTWO:PERcent?
```

Arguments <value> ::= <NRf> the limit in percent (%).

Examples
```
[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BLE:DELTA:FTWO:PERCENT?
```
might return 99.00 indicating the limit of the Low Energy ΔF2 percentage is 99%.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FTWO:RESult

Sets or returns the limit of the Low Energy ΔF2 result in the Bluetooth Constellation Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FTWO:RESult
<value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:DELTa:FTWO:RESult?
```

Arguments <value> ::= <NRf> the limit in Hz.

Examples
```
[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BLE:DELTA:FTWO:RESULT
```
185.0E+3 sets the limit of the Low Energy ΔF2 result to 185.0 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:DRIFt:FNFZero

Sets or returns the limit of the Low Energy maximum drift fn–f0 in the Bluetooth Constellation Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:DRIFt:FNFZero
<value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:DRIFt:FNFZero?
```

Arguments <value> ::= <NRf> the limit in Hz.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BLE:MAX:DRIFT:FNFZERO?` might return 50.0E+3 indicating the limit of the Low Energy maximum drift fn–f0 is 50 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:FREQuency:OFFSet

Sets or returns the limit of the Low Energy maximum frequency offset in the Bluetooth Constellation Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:FREQuency:OFFSet
<value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:FREQuency:OFFSet?
```

Arguments <value> ::= <NRf> the limit in Hz.

Examples

```
[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BLE:MAX:FREQUENCY:OFFSET
```

75.0E+3 sets the Basic Rate maximum frequency offset limit to 75 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:FREQuency:OFFSet:PREAmble

Sets or returns the limit of the Low Energy maximum frequency offset value from the preamble region in the Bluetooth Constellation Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax

```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:FREQuency:OFFSet:
PREAmble <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:MAX:FREQuency:OFFSet:
PREAmble?
```

Arguments <value> ::= <NRf> the limit in Hz.

Examples

```
[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BLE:MAX:FREQUENCY:OFFSET:
PREAMBLE
```

75.0E+3 sets the limit of the Low Enegy maximum frequency offset limit to 75 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:AVERage:MAX

Sets or returns the limit of the Low Energy maximum average output power value in the Bluetooth Constellation Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:AVERage:MAX
<value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:AVERage:MAX?
```

Arguments <value> ::= <NRf> the limit in dBm.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BLE:OPOWER:AVERAGE:MAX?` might return 10.00 indicating the Low Energy maximum output power average is 10.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:AVERage:MIN

Sets or returns the limit of the Low Energy minimum average output power value in the Bluetooth Constellation Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:AVERage:MIN
<value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:AVERage:MIN?
```

Arguments <value> ::= <NRf> the limit in dBm.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BLE:OPOWER:AVERAGE:MIN?` might return -20.00 indicating the Low Energy minimum output power average is -20 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:PEAK:MAX

Sets or returns the limit of the Low Energy maximum peak output power value in the Bluetooth Constellation Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:PEAK:MAX <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:PEAK:MAX?
```

Arguments <value> ::= <NRf> the limit in dBm.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BLE:OPOWER:PEAK:MAX?` might return 13.00 indicating the maximum peak output power is 13.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:PEAK:MIN

Sets or returns the limit of the Low Energy minimum peak output power value in the Bluetooth Constellation Summary display.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:PEAK:MIN <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:BLE:OPOWer:PEAK:MIN?
```

Arguments <value> ::= <NRf> the limit in dBm.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:BLE:OPOWER:PEAK:MIN?` might return -17.00 indicating the maximum peak output power is -17.00 dBm.

[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:F1FZero

Sets or returns the limit of the maximum fn–fn0 drift in the packet from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:F1FZero <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:F1FZero?
```

Arguments <value> ::= <NRf> the limit in Hz.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:MAX:DRIFT:F1FZERO` 20.0E+3 sets the maximum fn–fn0 drift in the packet to 20.00 kHz.

[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:FNFN5

Sets or returns the limit of the maximum fn–fn5 drift in the packet from the Bluetooth summary.

Conditions Measurement views: Bluetooth summary

Group Sense commands

Syntax
```
[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:FNFN5 <value>
[SENSe]:BLUEtooth:SUMMary:LIMIt:MAX:DRIFt:FNFN5?
```

Arguments <value> ::= <NRf> the limit in Hz.

Examples `[SENSE]:BLUETOOTH:SUMMARY:LIMIT:MAX:DRIFT:FNFN5` 20.0E+3 sets the maximum fn–fn5 drift in the packet to 20.00 kHz.

[SENSe]:BLUEtooth:SYMBol:POINts

Sets or returns the number of points per symbol (how many points to use between symbols) in the Bluetooth measurement.

Conditions Measurement views: Any Bluetooth display.

Group Sense commands

Syntax

```
[SENSe]:BLUEtooth:SYMBol:POINts { ONE |TWO | FOUR | EIGHt |
SIXTeen | THIRtytwo }
[SENSe]:BLUEtooth:SYMBol:POINts?
```

Arguments The number of points per symbol.

Examples `[SENSE]:BLUETOOTH:SYMBOL:POINTS` FOUR sets the number of points per symbol to four.

[SENSe]:BLUEtooth:TEST:PATTern

Sets or returns the test pattern type in the Bluetooth displays.

Conditions Measurement views: Any Bluetooth display.

Group Sense commands

Syntax

```
[SENSe]:BLUEtooth:TEST:PATTern
{LDEViation|HDEViation|TPOTHER}
[SENSe]:BLUEtooth:TEST:PATTern?
```

Related Commands [SENSe]:BLUEtooth:TEST:PATTern:AUTO:DETEct

Arguments LDEViation indicates the low deviation test pattern.

HDEViation indicates the high deviation test pattern.

TPOTHER indicates a user defined test pattern.

Examples `[SENSE]:BLUETOOTH:TEST:PATTERN`? might return LDEV indicating the selected test pattern type is the low deviation type.

[SENSe]:BLUEtooth:TEST:PATTern:AUTO:DETEct

Sets or returns the auto detect setting for the Bluetooth displays.

Conditions Measurement views: Any Bluetooth display.

Group Sense commands

Syntax

```
[SENSe]:BLUEtooth:TEST:PATTern:AUTO:DETEct {0|1}
[SENSe]:BLUEtooth:TEST:PATTern:AUTO:DETEct?
```

Related Commands [SENSe]:BLUEtooth:TEST:PATTern

Arguments 1 indicates that the auto detect test pattern is enabled (checked in the view). 0 indicates that the auto detect test pattern is not enabled (not checked).

Examples `[SENSE]:BLUETOOTH:TEST:PATTERN:AUTO:DETECT?` might return 0 indicating that the auto detect test pattern is not enabled.

[SENSe]:BLUEtooth:TIME:ZERO:REFERENCE

Sets or returns the Time Zero Reference setting in the Bluetooth display.

Conditions Measurement views: Any Bluetooth display.

Group Sense commands

Syntax

```
[SENSe]:BLUEtooth:TIME:ZERO:REFERENCE {Acquisistion
Start|Trigger}
[SENSe]:BLUEtooth:TIME:ZERO:REFERENCE?
```

Arguments Acquisition Start sets the Time Zero Reference to the start of the acquisition.

Trigger sets the Time Zero Reference to the trigger position.

Examples `[SENSE]:BLUETOOTH:TIME:ZERO:REFERENCE?` might return TRIGGER indicating the Time Zero Reference is the trigger position.

[SENSe]:BLUEtooth:TRACe:TYPE

Sets or queries the type of trace on the Trace tab of the control panel.

Conditions Measurement views: All

Group Sense commands

Syntax

```
[SENSe]:BLUEtooth:TRACe:TYPE {IQ|FREQDEV}
[SENSe]:BLUEtooth:TRACe:TYPE?
```

Arguments

IQ specifies the IQ trace.

FREQDEV specifies the Frequency Deviation trace.

Examples

`[SENSE]:BLUETOOTH:TRACE:TYPE?` might return IQ indicating the Trace type is set to IQ.

[SENSe]:CCDF:{BANDwidth|BWIDth}

Sets or queries the CCDF measurement bandwidth (frequency span).

Conditions

Measurement views: CCDF

Group

Sense commands

Syntax

```
[SENSe]:CCDF:{BANDwidth|BWIDth} <value>
[SENSe]:CCDF:{BANDwidth|BWIDth}?
```

Arguments

`<value> :: <NRf>` is the CCDF measurement bandwidth.
Range: 10 Hz to 40 MHz (Standard) / 60 MHz (Option 110).

Examples

`SENSE:CCDF:BANDWIDTH 1MHz` sets the CCDF measurement bandwidth to 1 MHz.

[SENSe]:CCDF:CLEar (No Query Form)

Clears the CCDF accumulator and restarts the measurement.

Conditions

Measurement views: CCDF

Group

Sense commands

Syntax

```
[SENSe]:CCDF:CLEar
```

Arguments

None

Examples `SENSE:CCDF:CLEAR` clears the CCDF accumulator and restarts the measurement.

[SENSe]:CCDF:TIME:TOTal:LENGth

Sets or queries the CCDF measurement time when [SENSe]:CCDF:TIME:TYPE is set to TOTal.

Conditions Measurement views: CCDF

Group Sense commands

Syntax
```
[SENSe]:CCDF:TIME:TOTal:LENGth <value>
[SENSe]:CCDF:TIME:TOTal:LENGth?
```

Arguments `<value> ::= <NRf>` specifies the CCDF measurement time.
Range: 20 ms to 100 s.

Examples `SENSE:CCDF:TIME:TOTAL:LENGTH 10` sets the CCDF measurement time to 10 s.

[SENSe]:CCDF:TIME:TYPE

Determines how to repeat the CCDF measurement.

Conditions Measurement views: CCDF

Group Sense commands

Syntax
```
[SENSe]:CCDF:TIME:TYPE { SINGLe | TOTal | CONTinuous }
[SENSe]:CCDF:TIME:TYPE?
```

Related Commands INITiate commands

Arguments `SINGLe` specifies that the analyzer sets the analysis length to 1 ms and then acquire data once to calculate CCDF.

TOTal specifies that the analyzer sets the analysis length to 20 ms and then repeats data acquisition and CCDF calculation for the time specified by the [SENSe]:CCDF:TIME:TOTal:LENGth command.

CONTinuous specifies that the analyzer sets the analysis length to 1 ms and then repeats data acquisition and CCDF calculation continuously. To reset the process, use the [SENSe]:CCDF:CLEar command or the INITiate commands.

Examples

SENSE:CCDF:TIME:TYPE SINGLe specifies that the analyzer sets the analysis length to 1 ms and then acquire data once to calculate CCDF.

[SENSe]:DDEMod:ANALysis:LENGth

Sets or queries the analysis length in seconds or symbols. The command [SENSe]:DDEMod:TIMe:UNITs determines which is used. For example, if the symbol rate is 1 MHz and the acquisition length is 20 ms, the range may be 0 to 19999. Setting a specified length changes [SENSe]:DDEMod:ANALysis:AUTO to OFF.

Conditions

Measurement views: General purpose digital modulation

Group

Sense commands

Syntax

```
[SENSe]:DDEMod:ANALysis:LENGth <value>
[SENSe]:DDEMod:ANALysis:LENGth?
```

Related Commands

[SENSe]:DDEMod:ANALysis:LENGth:ACTual?

[SENSe]:DDEMod:TIMe:UNITs

[SENSe]:DDEMod:ANALysis:LENGth:AUTO

Arguments

<value> :: <NRf> specifies the analysis length in seconds or symbols.
Range in seconds: 200 ns to [(acquisition length) - 400 ns].

If [(analysis start) + (analysis length)] > [(acquisition length) - 400 ns], the actual analysis length is reduced to [(acquisition length) - 200 ns].
Range in symbols: 200 ns * (symbol rate) to [(acquisition length) - 400 ns] * (symbol rate).

If [(analysis start) + (analysis length)] > [(acquisition length) - 400 ns], the actual analysis length is reduced to [(acquisition length) - 200 ns] * (symbol rate).

Examples

`SENSE:DDEMOD:ANALYSIS:LENGTH 25.625us` sets the analysis length to 25.625 µs.

`SENSE:DDEMOD:ANALYSIS:LENGTH 256` sets the analysis length to 256 symbols.

[SENSe]:DDEMod:ANALysis:LENGth:ACTual? (Query Only)

Queries the actual analysis length which returns a value in either seconds or symbols. The command [SENSe]:DDEMod:TIMe:UNITs determines which is used.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax

```
[SENSe]:DDEMod:ANALysis:LENGth:ACTual?
```

Related Commands [SENSe]:DDEMod:ANALysis:LENGth

[SENSe]:DDEMod:TIMe:UNITs

Arguments None

Returns `<NRf>` Actual analysis length in seconds or symbols.

Examples `SENSE:DDEMOD:ANALYSIS:LENGTH:ACTUAL?` might return `25.625E-6`, indicating that the actual analysis length is 25.625 µs.

[SENSe]:DDEMod:ANALysis:LENGth:AUTO

Determines whether to set the analysis length automatically or manually.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax

```
[SENSe]:DDEMod:ANALysis:LENGth:AUTO { OFF | ON | 0 | 1 }
[SENSe]:DDEMod:ANALysis:LENGth:AUTO?
```

Arguments

`OFF` or `0` sets the analysis length manually, using the [SENSe]:DDEMod:ANALysis:LENGth command.

`ON` or `1` sets the analysis length automatically.

Examples

`SENSE:DDEMOD:ANALYSIS:LENGTH:AUTO ON` sets the analysis length automatically.

[SENSe]:DDEMod:{BANDwidth|BWIDth}:MODE

Sets or queries the measurement bandwidth mode for Digital Demod measurements.

Conditions

Measurement views: General purpose digital modulation

Group

Sense commands

Syntax

```
[SENSe]:DDEMod:{BANDwidth|BWIDth}:MODE { MANual | AUTO | LSPan }
[SENSe]:DDEMod:{BANDwidth|BWIDth}:MODE?
```

Related Commands

[SENSe]:DDEMod:{BANDwidth|BWIDth}:TINTerval

Arguments

`MANual` specifies manual bandwidth control mode. A typical use is setting a narrow measurement bandwidth on a small range of frequencies to examine one signal while rejecting the others that are present in a Spectrum graph.

`AUTO` specifies automatic bandwidth control mode where a measurement picks a bandwidth based on other parameter settings (such as symbol rate, modulation type, filter, and so on).

`LSPan` (LinkToSpan) specifies Lspan bandwidth control mode, where you can use the Spectrum display to tune the frequency, view the signal, and use the Span control to set the measurement bandwidth. This mode emulates legacy instruments.

Examples

`DDEMOD:BANDWIDTH:MODE AUTO` sets the analyzer bandwidth mode to automatically select a measurement bandwidth.

[SENSe]:DDEMod:{BANDwidth|BWIDth}:TINTerval

Sets or queries the measurement bandwidth (frequency span).

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax
```
[SENSe]:DDEMod:{BANDwidth|BWIDth}:TINTerval <value>
[SENSe]:DDEMod:{BANDwidth|BWIDth}:TINTerval?
```

Related Commands [SENSe]:DDEMod:SRATe

Arguments <value> :: <NRf> specifies the measurement bandwidth.
Range: Symbol rate to 40 MHz or, with Option 110, 110 MHz.

The minimum value depends on the setting of [:SENSe]:DDEMod:SRATe.

Returns <NRf> Actual measurement bandwidth.

Examples SENSe:DDEMOD:BANDWIDTH:TINTERVAL 35.255MHz sets the measurement bandwidth to 35.255 MHz.

[SENSe]:DDEMod:{BANDwidth|BWIDth}:TINTerval:AUTO

Sets or queries the measurement bandwidth (frequency span) automatically.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax
```
[SENSe]:DDEMod:{BANDwidth|BWIDth}:TINTerval:AUTO { OFF | ON
| 0 | 1 }
[SENSe]:DDEMod:{BANDwidth|BWIDth}:TINTerval:AUTO?
```

Related Commands [SENSe]:DDEMod:{BANDwidth|BWIDth}:TINTerval

Arguments OFF or 0 sets the measurement bandwidth manually, using the [SENSe]:DDEMod:{BANDwidth|BWIDth}:TINTerval command.

ON or 1 sets the measurement bandwidth automatically.

Examples `SENSE:DDEMOD:BANDWIDTH:TINTERVAL:AUTO` sets the measurement bandwidth automatically.

[SENSe]:DDEMod:BURSt:DETect

Sets or queries burst detection.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax
```
[SENSe]:DDEMod:BURSt:DETect { ON | OFF }
[SENSe]:DDEMod:BURSt:DETect?
```

Related Commands [SENSe]:DDEMod:BURSt:THReshold

Arguments `ON` analyzes just that burst period if a burst is found. If a burst is not found, does not analyze but displays an error message.

`OFF` analyzes the whole analysis length.

NOTE. *When selecting On and if the signal is not adequate for the demodulation, the measurement will fail and show an error message.*

Examples `SENSE:DDEMOD:BURST:DETECT OFF` disables burst detection and analyzes the whole analysis length.

[SENSe]:DDEMod:BURSt:THReshold

Sets or queries the threshold level above which the input signal is determined to be a burst.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax
```
[SENSe]:DDEMod:BURSt:THReshold <value>
[SENSe]:DDEMod:BURSt:THReshold?
```

Related Commands [SENSe]:DDEMod:BURSt:DETect

Arguments `<value> :: <NRf>` specifies the threshold level for detecting bursts.
Range: -100 to -10 dBc.

Examples `SENSE:DDEMOD:BURST:THRESHOLD -25` sets the threshold level to -25 dBc.

[SENSe]:DDEMod:CARRier:OFFSet

Sets or queries the carrier frequency offset in digital modulation analysis.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax
```
[SENSe]:DDEMod:CARRier:OFFSet <value>
[SENSe]:DDEMod:CARRier:OFFSet?
```

Related Commands The settings of the following commands may narrow the effective range.

[SENSe]:DDEMod:FILTer:ALPHa

[SENSe]:DDEMod:FILTer:REFerence

[SENSe]:DDEMod:MODulation:TYPE

[SENSe]:DDEMod:SRATe

Arguments `<value> :: <NRf>` specifies the carrier frequency offset.
Range: -20 MHz to +20 MHz or with Option 110, -55 MHz to +55 MHz .

Examples `SENSE:DDEMOD:CARRIER:OFFSET 2kHz` sets the carrier frequency offset to 2 kHz.

[SENSe]:DDEMod:CARRier:OFFSet:AUTO

Sets or queries whether to detect the carrier frequency automatically in digital modulation analysis.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax
```
[SENSe]:DDEMod:CARRier:OFFSet:AUTO { OFF | ON | 0 | 1 }
[SENSe]:DDEMod:CARRier:OFFSet:AUTO?
```

Related Commands [SENSe]:DDEMod:CARRier:OFFSet

Arguments `OFF` or `0` sets the carrier frequency manually, using the[SENSe]:DDEMod:CARRier:OFFSet command.

`ON` or `1` detects the carrier frequency automatically.

Examples `[SENSE]:DDEMOD:CARRIER:OFFSET:AUTO` ON enables automatic detection of the carrier frequency.

[SENSe]:DDEMod:EQUalizer:AVAilable? (Query Only)

Determines if the equalizer is available.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax
```
[SENSe]:DDEMod:EQUalizer:AVAilable?
```

Arguments None

Returns 0 or 1

Where:

`0` indicates the equalizer is not available for the current modulation type and reference filter.

`1` indicates the equalizer is available.

[SENSe]:DDEMod:EQUalizer:CONVergence

Sets or queries the Convergence value (also known as tap update rate).

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax
```
[SENSe]:DDEMod:EQUalizer:CONVergence <value>
[SENSe]:DDEMod:EQUalizer:CONVergence?
```

Arguments <value> :: <NR3> specifies the convergence value.

Examples SENSE:DDEMOD:EQUALIZER:CONVERGENCE? might return 1.0000000000E-6.

[SENSe]:DDEMod:EQUalizer:ENABle

Sets or queries the Equalizer state.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax
```
[SENSe]:DDEMod:EQUalizer:ENABle {OFF | ON | 0 | 1 }
[SENSe]:DDEMod:EQUalizer:ENABle?
```

Related Commands [SENSe]:DDEMod:EQUalizer:MODE

Arguments ON or 1 enables the Equalizer.

OFF or 0 disables the Equalizer.

Examples SENS:DDEM:EQU:ENAB 1 might return 1, indicating that the equalizer is enabled.

[SENSe]:DDEMod:EQUalizer:LENGth

Sets or queries the equalizer filter length.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax `[SENSe]:DDEMod:EQUalizer:LENGth <value>`

Arguments `<value> ::` `<NR1>` specifies the equalizer filter length.

Examples `SENSE:DDEMOD:EQUALIZER:LENGTH 39` sets the filter length to 39 symbols.

[SENSe]:DDEMod:EQUalizer:MODE

Sets or queries the equalizer mode. The equalizer can in learning mode (train) or enabled (hold).

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax
```
[SENSe]:DDEMod:EQUalizer:MODE {TRAIn | HOLD }
[SENSe]:DDEMod:EQUalizer:MODE?
```

Related Commands [SENSe]:DDEMod:EQUalizer:ENABle

Arguments `TRAIn` places the equalizer in training mode.

`HOLD` places in the equalizer in enabled mode.

Examples `SENSE:DDEMOD:EQUALIZER:MODE HOLD` places the equalizer in enabled mode.

[SENSe]:DDEMod:EQUalizer:RESet (No Query Form)

Resets the equalizer settings to default values.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax `[SENSe]:DDEMod:EQUalizer:RESet`

Examples `SENSE:DDEMOD:EQUALIZER:RESET` sets the equalizer parameters to their default values.

[SENSe]:DDEMod:EQUalizer:TAPS

Sets or queries the number of filter coefficients for the equalizer.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax
```
[SENSe]:DDEMod:EQUalizer:TAPS <value>
[SENSe]:DDEMod:EQUalizer:TAPS?
```

Arguments <value> ::= <NR1> the filter coefficient for the equalizer.

Examples `[SENSE]:DDEMOD:EQUALIZER:TAPS 65` sets the filter coefficient to 65.

[SENSe]:DDEMod:EQUalizer:TSRAtio

Sets or queries the Equalizer Taps/Symbol parameter.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax
```
[SENSe]:DDEMod:EQUalizer:TSRAtio { ONE | TWO | FOUR | EIGHt
}
```

Arguments Valid arguments are ONE, TWO, FOUR, and EIGHt. Numerals are not valid arguments.

Examples `SENSE:DDEMOD:EQUALIZER:TSRATIO FOUR` sets the Taps/Symbol value to four.

[SENSe]:DDEMod:FILTer:ALPHa

Sets or queries the filter factor (α/BT) in the digital modulation analysis.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax

```
[SENSe]:DDEMod:FILTer:ALPHa <value>
[SENSe]:DDEMod:FILTer:ALPHa?
```

Arguments <value> :: <NRf> specifies the filter factor. Range: 0.001 to 1.

Examples SENSE:DDEMOD:FILTER:ALPHA 0.5 sets the filter factor to 0.5.

[SENSe]:DDEMod:FILTer:MEASurement

Sets or queries the measurement filter in the digital modulation analysis.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax

```
[SENSe]:DDEMod:FILTer:MEASurement { OFF | RRCosine | RCOSine
| GAUSsian | RECTangular | IS95TXEQ_MEA | IS95TX_MEA |
C4FM_P25 | USERx | UOTHer }
[SENSe]:DDEMod:FILTer:MEASurement?
```

Arguments The following table lists the arguments.

Table 2-29: Digital modulation measurement filter

Argument	Measurement filter
OFF	No filter
RRCosine	Root Raised Cosine
RCOSine	Raised Cosine
GAUSsian	Gaussian
RECTangular	Rectangular

Table 2-29: Digital modulation measurement filter (cont.)

Argument	Measurement filter
IS95TXEQ_MEA	IS95 receive filter for the transmitter configured with both the transmit filter and the phase equalizer.
IS95TX_MEA	IS95 receive filter for the transmitter configured with only the transmit filter.
C4FM_P25	C4FM-P25
USER1	User defined Measurement Filter 1
USER2	User defined Measurement Filter 2
USER3	User defined Measurement Filter 3
UOTHer	Other user defined Measurement Filter

Examples

`SENSE:DDEMOD:FILTER:MEASUREMENT RRCosine` selects the Root Raised Cosine filter as the measurement filter.

[SENSe]:DDEMod:FILTer:REFerence

Sets or queries the reference filter in the digital modulation analysis.

Conditions

Measurement views: General purpose digital modulation

Group

Sense commands

Syntax

```
[SENSe]:DDEMod:FILTer:REFerence { OFF | RCOSine | GAUSsian |
RECTangular  | IS95REF | HSINe | SOQPSK_MIL | SOQPSK_ARTM |
SBPSK_MIL | USERx | UOTHer }
[SENSe]:DDEMod:FILTer:REFerence?
```

Arguments

The following table lists the arguments.

Table 2-30: Digital modulation reference filter

Argument	Measurement filter
OFF	No filter
RCOSine	Raised Cosine
GAUSsian	Gaussian
RECTangular	Rectangular
IS95REF	IS95 reference filter including the response of the transmit filter, the phase equalizer, and the receive (complementary) filter.
HSINe	Half Sine

Table 2-30: Digital modulation reference filter (cont.)

Argument	Measurement filter
SOQPSK_MIL	SOQPSK–MIL
SOQPSK_ARTM	SOQPSK–ARTM
SBPSK_MIL	SBPSK–MIL
USER1	User defined Measurement Filter 1
USER2	User defined Measurement Filter 2
USER3	User defined Measurement Filter 3
UOTHer	Other user defined Measurement Filter

Examples `SENSE:DDEMOD:FILTER:REFERENCE RCOSine` selects the Raised Cosine filter as the reference filter.

[SENSe]:DDEMod:FREQuency:DEViation

Sets or queries the frequency deviation in the digital modulation analysis. Programming a specified frequency deviation sets [SENSe]:DDEMod:FREQuency:DEViation:AUTO OFF.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to C4FM, FSK2, FSK4, FSK8, or FSK16.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax

```
[SENSe]:DDEMod:FREQuency:DEViation <value>
[SENSe]:DDEMod:FREQuency:DEViation?
```

Arguments `<value> ::  <NRf>` sets the frequency deviation.

Standard product range: 1 Hz to 40 MHz.

Option 110 product range: 1 Hz to 110 MHz.

Examples `SENSE:DDEMOD:FREQUENCY:DEVIATION 1MHz` sets the frequency deviation to 1 MHz.

[SENSe]:DDEMod:FREQuency:DEViation:AUTO

Determines whether to detect automatically or set manually the frequency deviation used to determine the symbol values of an FSK or C4FM signal.

This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to C4FM, FSK2, FSK4, FSK8, or FSK16.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax

```
[SENSe]:DDEMod:FREQuency:DEViation:AUTO { OFF | ON | 0 | 1 }
[SENSe]:DDEMod:FREQuency:DEViation:AUTO?
```

Arguments `ON` or `1` automatically calculates the frequency deviation for the analysis range (default).

`OFF` or `0` sets the frequency deviation using the [SENSe]:DDEMod:FREQuency:DEViation command.

Examples `SENSE:DDEMOD:FREQUENCY:DEVIATION:AUTO ON`automatically calculates the frequency deviation.

[SENSe]:DDEMod:MAGNitude:NORMalize

Sets or queries the method for the magnitude normalization.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax

```
[SENSe]:DDEMod:MAGNitude:NORMalize { RSYMbol | MSYMbol }
[SENSe]:DDEMod:MAGNitude:NORMalize?
```

Arguments `RSYMbol` normalizes the magnitude with the RMS symbol magnitude.

`MSYMbol` normalizes the magnitude with the maximum symbol magnitude.

Examples `SENSE:DDEMOD:MAGNITUDE:NORMALIZE RSYMbol` normalizes the magnitude with the RMS symbol magnitude.

[SENSe]:DDEMod:MINDex

Sets or queries the modulation index of a CPM signal. This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to CPM and [SENSe]:DDEMod:MINDex:AUTO is set to OFF.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax

```
[SENSe]:DDEMod:MINDex { 1 | 2 | 3 | 4 | 5 | 6 }
[SENSe]:DDEMod:MINDex?
```

Arguments The following table shows the arguments and modulation index.

CPM modulation index

Argument	Modulation index
1	4/16, 5/16
2	5/16, 6/16
3	6/16, 7/16
4	7/16, 10/16
5	12/16, 13/16
6	8/16, 8/16

Examples `SENSE:DDEMOD:MINDEX 1` selects the modulation index to “4/16, 5/16”.

[SENSe]:DDEMod:MINDex:AUTO

Determines whether to detect automatically or set manually the modulation index of a CPM signal. This command is valid when [SENSe]:DDEMod:MODulation:TYPE is set to CPM.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax

```
[SENSe]:DDEMod:MINDex:AUTO { OFF | ON | 0 | 1 }
[SENSe]:DDEMod:MINDex:AUTO
```

Arguments `ON` or `1` automatically calculates the modulation index for the analysis range.

`OFF` or `0` sets the modulation index using the [SENSe]:DDEMod:MINDex command.

Examples `SENSE:DDEMOD:MINDEX:AUTO ON` automatically calculates the modulation index for the analysis range.

[SENSe]:DDEMod:MODulation:TYPE

Sets or queries the modulation type in the digital modulation analysis.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax

```
[SENSe]:DDEMod:MODulation:TYPE { QPSK | PSK8 | D8PSK |
PIOVER2DBPSK | DQPSK | PIOVER4DQPSK | BPSK | OQPSK | QAM16 |
QAM32 | QAM64 | QAM128 | QAM256 | MSK | FSK2 | FSK4 | FSK8 |
FSK16 | CPM | APSK16| APSK32 | SOQPSK | SBPSK | C4FM }
[SENSe]:DDEMod:MODulation:TYPE?
```

Arguments The following table lists the arguments and corresponding modulation type.

Table 2-31: Modulation type

Argument	Modulation type
QPSK	QPSK
PSK8	8PSK
D8PSK	D8PSK
PIOVER2DBPSK	π/2 DBPSK
DQPSK	DQPSK
PIOVER4DQPSK	π/4 QPSK
BPSK	BPSK
OQPSK	OQPSK
QAM16	16QAM
QAM32	32QAM
QAM64	64QAM
QAM128	128QAM
QAM256	256QAM
MSK	MSK

Table 2-31: Modulation type (cont.)

Argument	Modulation type
FSK2	FSK2
FSK4	FSK4
FSK8	FSK8
FSK16	FSK16
CPM	CPM
APSK16	APSK16
APSK32	APSK32
SOQPSK	SOQPSK
SBPSK	SBPSK
C4FM	C4FM

Examples `SENSE:DDEMOD:MODULATION:TYPE QPSK` selects QPSK modulation system.

[SENSe]:DDEMod:PRESet (No Query Form)

Presets the modulation analysis to a communication standard.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax `[SENSe]:DDEMod:PRESet <standard_name>`

Related Commands [SENSe]:DDEMod:MODulation:TYPE

Arguments The following table lists the preset standard names with their modulation types and settings.

Table 2-32: Modulation Presets

Preset Name	Modulation type	Settings
“802.15.4”	OQPSK	Symbol rate: 1 MHz
		Meas Filter: None
		Ref Filter: Half sine
		Filter Param: None
		Symbol Pt Location: NA
		Remove Q offset: not checked
“SBPSK-MIL”	SBPSK	Symbol rate: 2.4 kHz
		Meas Filter: None
		Ref Filter: SBPSK-MIL
		Filter Param: 0.5
		Symbol Pt Location: NA
		Remove Q offset: NA
“SOQPSK-MIL”	SOQPSK	Symbol rate: 2.4 kHz
		Meas Filter: None
		Ref Filter: SOQPSK-MIL
		Filter Param: 0.5
		Symbol Pt Location: Center
		Remove Q offset: checked
“CPM-MIL”	CPM	Symbol rate: 19.2 KHz
		Meas Filter: None
		Ref Filter: None
		Filter Param: None
		Symbol Pt Location: NA
		Remove Q offset: NA
“SOQPSK-ARTM Tier 1”	SOQPSK	Symbol rate: 2.5 MHz
		Meas Filter: None
		Ref Filter: SOQPSK-ARTM
		Filter Param: None
		Symbol Pt Location: Center
		Remove Q offset: checked
“Project25 Phase I”	C4FM	Symbol rate: 4.8 kHz
		Meas Filter: C4FM-P25
		Ref Filter: Raised cosine
		Filter Param: 0.2
		Symbol Pt Location: NA
		Remove Q offset: NA

Table 2-32: Modulation Presets (cont.)

Preset Name	Modulation type	Settings
"CDMA2000-Base"	QPSK	Symbol rate: 1.2288 MHz
		Meas Filter: IS95 TXEQ_MEA
		Ref Filter: IS95 REF
		Filter Param: None
		Symbol Pt Location: NA
		Remove Q offset: NA
"W-CDMA"	QPSK	Symbol rate: 3.84 MHz
		Meas Filter: Root raised cosine
		Ref Filter: Raised cosine
		Filter Param: 0.22
		Symbol Pt Location: NA
		Remove Q offset: NA

In addition to the specific settings listed in the table, the following general settings are also made when you load any of the defined presets.

Setting	Value
Points/symbol	4
Burst detection mode	Off
Burst detection threshold	-10 dBc
Analysis offset	Auto
Analysis length	Auto
Frequency offset	Auto
Measurement BW	Auto
Frequency deviation	Auto
Modulation index	Auto

Examples `[SENSE]:DDEMOD:PRESET "802.15.4"` sets demodulation to the standard OQPSK modulation.

[SENSe]:DDEMod:RING:RADius

Sets or queries the ring radius in the digital APSK modulation analysis.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax
```
[SENSe]:DDEMod:RING:RADius <value>
[SENSe]:DDEMod:RING:RADius?
```

Arguments <value> :: <NRf> specifies the ring radius.

APSK16 valid ring arguments are 1 and 2.

APSK32 valid ring arguments are 1, 2, and 3.

Examples
```
[SENSE]:DDEMOD:RING:RADIUS
```

[SENSe]:DDEMod:RING:ROTation

Sets or queries the ring rotation in the digital APSK modulation analysis.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax
```
[SENSe]:DDEMod:RING:ROTation <value>
[SENSe]:DDEMod:RING:ROTation?
```

Arguments <value> :: <NRf> specifies the ring rotation.

APSK16 valid ring arguments are 1 and 2.

APSK32 valid ring arguments are 1, 2, and 3.

Examples
```
[SENSE]:DDEMOD:RING:ROTATION
```

[SENSe]:DDEMod:SRATe

Sets or queries the symbol rate in the digital modulation analysis.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax

```
[SENSe]:DDEMod:SRATe <value>
[SENSe]:DDEMod:SRATe?
```

Arguments

<value> :: <NRf> specifies the symbol rate.

Standard product range: 100 Hz to 40 MHz.

Option 110 product range: 100 Hz to 110 MHz.

Examples

SENSE:DDEMOD:SRATE 21.0E3 sets the symbol rate to 21 kHz.

[SENSe]:DDEMod:SWAP:IQ

Determines whether or not to exchange I and Q data before demodulating.

Conditions

Measurement views: General purpose digital modulation

Group

Sense commands

Syntax

```
[SENSe]:DDEMod:SWAP:IQ { OFF | ON | 0 | 1 }
[SENSe]:DDEMod:SWAP:IQ?
```

Arguments

OFF or 0 uses I and Q data as they are.

ON or 1 exchanges I and Q data.

Examples

SENSE:DDEMOD:SWAP:IQ ON exchanges I and Q data before demodulating.

[SENSe]:DDEMod:SYMBol:HSSHift

Sets or queries the Q data half-symbol shift for OQPSK and SOQPSK signals.

Conditions

Measurement views: General purpose digital modulation

Group

Sense commands

Syntax

```
[SENSe]:DDEMod:SYMBol:HSSHift { OFF | ON | 0 | 1 }
[SENSe]:DDEMod:SYMBol:HSSHift?
```

Related Commands [SENSe]:DDEMod:MODulation:TYPE

Arguments `OFF` or `0` the Q offset shift is not applied.

`ON` or `1` the Q offset shift is applied.

Examples `DDEMod:SYMBol:HSSHift ON` the Q offset is applied.

[SENSe]:DDEMod:SYMBol:MAP:SOURce? (Query Only)

Queries the user symbol map.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax `[SENSe]:DDEMod:SYMBol:MAP:SOURce?`

Arguments None.

Examples `SENSE:DDEMOD:SYMBOL:MAP:SOURCE?` returns the symbol map filename.

[SENSe]:DDEMod:SYMBol:MAP[:STATe]

Determines whether or not to use the user symbol map.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax
```
[SENSe]:DDEMod:SYMBol:MAP[:STATe] { OFF | ON | 0 | 1 }
[SENSe]:DDEMod:SYMBol:MAP[:STATe]?
```

Related Commands [SENSe]:DDEMod:SYMBol:MAP:SOURce?

Arguments `OFF` or `0` disables the user symbol map.

`ON` or `1` enables the user symbol map.

Examples `SENSE:DDEMOD:SYMBOL:MAP:STATE ON` enables the user symbol map.

[SENSe]:DDEMod:SYMBol:PLOT:POSition

Sets or queries the symbol point location on an SOQPSK waveform.

Conditions Effective only on an SOQPSK waveform and on any general purpose digital demodulation views except Signal Quality and Symbol Table.

Group Sense commands

Syntax

```
[SENSe]:DDEMod:SYMBol:PLOT:POSition { EDGe | MIDDle }
[SENSe]:DDEMod:SYMBol:PLOT:POSition?
```

Related Commands

Arguments EDGe sets the symbol location at the edge of the eye opening.

MIDDle sets the symbol location at the middle of the eye opening.

Examples `[SENSE]:DDEMOD:SYMBOL:PLOT:POSITION` EDGE sets the symbol location the edge of the eye opening.

[SENSe]:DDEMod:SYMBol:POINts

Sets or queries the number of points per symbol (how many points to use between symbols when connecting the dots).

NOTE. *1 is not valid for the GMSK modulation.*

In the constellation view, select VECTors using the TRACe<x>:CONSte:MODE command first to change Points/Symbol.

Conditions Measurement views: Constellation, EVM versus Time, Magnitude error versus Time, Phase error versus Time, Freq Dev versus Time, Demod I&Q versus Time.

Group Sense commands

Syntax
```
[SENSe]:DDEMod:SYMBol:POINts { ONE | TWO | FOUR | EIGHt }
[SENSe]:DDEMod:SYMBol:POINts?
```

Arguments ONE, TWO, FOUR, and EIGHt represent the number of points per symbol.

The following table lists the conditions of use.

Argument	Modulation type supported
ONE	All but SOQPSK, OQPSK, SBPSK
TWO	All but SOQPSK
FOUR	All
EIGHt	All

Examples SENSE:DDEMOD:SYMBOL:POINTS FOUR sets the number of points per symbol to four.

[SENSe]:DDEMod:SYMBol:RATE:SEARch

Determines whether to enable a symbol rate search. This command is valid when the demodulation type is set to FSK2, FSK4, FSK8 or FSK16.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax
```
[SENSe]:DDEMod:SYMBol:RATE:SEARch { OFF | ON | 0 | 1 }
```

Related Commands [SENSe]:DDEMod:MODulation:TYPE

Arguments OFF or 0 disables the search.

ON or 1 enables the search.

Examples [SENSE]:DDEMOD:SYMBOL:RATE:SEARCH ON enables the symbol rate search.

[SENSe]:DDEMod:SYNCh:WORD

Determines whether to enable the synchronization word.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax
```
[SENSe]:DDEMod:SYNCh:WORD { OFF | ON | 0 | 1 }
[SENSe]:DDEMod:SYNCh:WORD?
```

Related Commands [SENSe]:DDEMod:SYNCh:WORD:SYMBol

Arguments `OFF` or `0` disables the synchronization word.

`ON` or `1` enables the synchronization word.

Examples `SENSE:DDEMOD:SYNCH:WORD ON` enables the synchronization word.

[SENSe]:DDEMod:SYNCh:WORD:SYMBol

Sets or queries the synchronization word when [SENSe]:DDEMod:SYNCh:WORD is ON. The word depends on the modulation type selected by the [SENSe]:DDEMod:MODulation:TYPE command.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax
```
[SENSe]:DDEMod:SYNCh:WORD:SYMBol <block>
[SENSe]:DDEMod:SYNCh:WORD:SYMBol?
```

Arguments `<block>::=#<num_digit><num_byte><sym(1)><sym(2)>...<sym(n)>`

Where
`<num_digit>` is the number of digits in <num_byte>.
`<num_byte>` is the number of bytes of data that follow.
`<sym(n)>` is the nth symbol value of the sync word. 32-bit integer.
n: Max 256.

Examples `SENSE:DDEMOD:SYNCH:WORD:SYMBOL #216xxxx` (4 symbols) sets a sync word composed of four symbols.

[SENSe]:DDEMod:TIMe:UNITs

Sets or queries the fundamental unit of time.

Conditions Measurement views: General purpose digital modulation

Group Sense commands

Syntax
```
[SENSe]:DDEMod:TIMe:UNITs { SEConds | SYMBols }
[SENSe]:DDEMod:TIMe:UNITs?
```

Arguments `SEConds` specifies the fundamental unit of time as seconds.

`SYMBols` specifies the fundamental unit of time as symbols.

Examples `SENSE:DDEMOD:TIME:UNITS SEConds` specifies the fundamental unit of time as seconds.

[SENSe]:DPX:AUDio:DEMod:GAIN

Sets or queries the audio gain.

NOTE. *The sound level is also affected by the Windows volume control.*

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax
```
[SENSe]:DPX:AUDio:DEMod:GAIN <value>
[SENSe]:DPX:AUDio:DEMod:GAIN?
```

Arguments `<value> ::= <NR1>` specifies the audio gain. Range: 0 to 15 (integer).

Examples `SENSE:DPX:AUDIO:DEMOD:GAIN 7` sets the audio gain to 7.

[SENSe]:DPX:AUDio:DEMod:RXBWidth

Sets or queries the receiver bandwidth in the audio demodulation.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax
```
[SENSe]:DPX:AUDio:DEMod:RXBWidth <value>
[SENSe]:DPX:AUDio:DEMod:RXBWidth?
```

Arguments `<value> ::` `<NRf>` specifies the receiver bandwidth in the audio demodulation.
Range: 1 kHz to 500 kHz.

Examples `SENSE:DPX:AUDIO:DEMOD:RXBWIDTH 30kHz` sets the receiver bandwidth to 30 kHz.

[SENSe]:DPX:AUDio:DEMod:RXFRequency? (Query Only)

Returns the receiver frequency in the audio demodulation. The frequency depends on the setting of the [SENSe]:DPX:AUDio:DEMod:TUNE command.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax
```
[SENSe]:DPX:AUDio:DEMod:RXFRequency?
```

Arguments None

Returns `<frequency> ::` `<NRf>` is the receiver frequency in the audio demodulation.

Examples `SENSE:DPX:AUDIO:DEMOD:RXFREQUENCY?` might return `80.3E+6`, indicating that the receiver frequency is 80.3 MHz.

[SENSe]:DPX:AUDio:DEMod:STATe

Determines whether to enable the audio demodulation.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax

```
[SENSe]:DPX:AUDio:DEMod:STATe { OFF | ON | 0 | 1 }
[SENSe]:DPX:AUDio:DEMod:STATe?
```

Arguments `OFF` or `0` disables the audio demodulation.

`ON` or `1` enables the audio demodulation.

Examples `SENSE:DPX:AUDIO:DEMOD:STATE ON` enables the audio demodulation.

[SENSe]:DPX:AUDio:DEMod:TUNE

Sets or queries how to determine the tuning frequency in the audio demodulation.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax

```
[SENSe]:DPX:AUDio:DEMod:TUNE { MR | MARK1 | MARK2 | MARK3|
MARK4 | SMARker | FREQcontrol }
[SENSe]:DPX:AUDio:DEMod:TUNE?
```

Arguments The following table lists the arguments.

Table 2-33: Frequency tuning

Argument	Tune with
MR	Reference marker (MR)
MARK1	Marker 1 (M1)
MARK2	Marker 2 (M2)
MARK3	Marker 3 (M3)
MARK4	Marker 4 (M4)

Table 2-33: Frequency tuning (cont.)

Argument	Tune with
SMARker	Selected marker
FREQcontrol	Center frequency setting

Examples `SENSE:DPX:AUDIO:DEMOD:TUNE MARK1` sets the tuning frequency to the value at Marker 1.

[SENSe]:DPX:AUDio:DEMod:TYPE

Sets or queries the modulation type in the audio demodulation.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax
```
[SENSe]:DPX:AUDio:DEMod:TYPE { AM | FM }
[SENSe]:DPX:AUDio:DEMod:TYPE?
```

Arguments `AM` selects the AM (Amplitude Modulation).

`FM` selects the FM (Frequency Modulation).

Examples `SENSE:DPX:AUDIO:DEMOD:TYPE FM` selects FM in the audio demodulation.

[SENSe]:DPX:{BANDwidth|BWIDth}:ACTual? (Query Only)

Queries the actual bandwidth whether set automatically or manually.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax
```
[SENSe]:DPX:{BANDwidth|BWIDth}:ACTual?
```

Arguments None

Examples `SENSE:DPX:BANDWIDTH:ACTUAL?` returns the RBW as a percentage of the Span.

[SENSe]:DPX:{BANDwidth|BWIDth}:RESolution

Sets or queries the resolution bandwidth (RBW) in the DPX, Spectrum measurement.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax

```
[SENSe]:DPX:{BANDwidth|BWIDth}:RESolution <value>
[SENSe]:DPX:{BANDwidth|BWIDth}:RESolution?
```

Arguments `<value> ::  <NRf>` specifies the resolution bandwidth.

Range: 1 Hz to 10 MHz.

Examples `SENSE:DPX:BANDWIDTH:RESOLUTION?` might return `100.00003E+3` for the RBW..

[SENSe]:DPX:{BANDwidth|BWIDth}[:RESolution]:AUTO

Determines whether to set the resolution bandwidth (RBW) automatically or manually in the DPX, Spectrum measurement.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax

```
[SENSe]:DPX:{BANDwidth|BWIDth}[:RESolution]:AUTO { OFF | ON
| 0 | 1 }
[SENSe]:DPX:{BANDwidth|BWIDth}[:RESolution]:AUTO?
```

Arguments `OFF` or `0` specifies that the resolution bandwidth is set manually using the [SENSe]:DPX:{BANDwidth|BWIDth}:ACTual? command.

`ON` or `1` specifies that the resolution bandwidth is set automatically.

Examples `SENSE:DPX:BANDWIDTH:AUTO ON` sets the resolution bandwidth automatically.

[SENSe]:DPX:CLEar:RESults (No Query Form)

Restarts multi-trace functions (Average and Max/Min Hold).

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax `[SENSe]:DPX:CLEar:RESults`

Arguments None

Examples `SENSE:DPX:CLEAR:RESULTS` restarts multi-trace functions.

[SENSe]:DPX:COLor

Sets or queries the color palette of three-dimensional graphs.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax

```
[SENSe]:DPX:COLor { RED | GREen | BLUe | CYAN | BCYan |
YELLow | MAGenta | GRAY | TEMPerature | SPECtral }
[SENSe]:DPX:COLor?
```

Arguments The following table lists the arguments.

Table 2-34: Color palette for DPX, Spectrum

Argument	Palette
RED	Red
GREen	Green
BLUe	Blue
CYAN	Cyan
BCYan	Binary cyan

Table 2-34: Color palette for DPX, Spectrum (cont.)

Argument	Palette
YELLow	Yellow
MAGenta	Magenta
GRAY	Gray
TEMPerature	Temperature
SPECtral	Spectral

Examples `SENSE:DPX:COLOR TEMPERATURE` selects the temperature color palette.

[SENSe]:DPX:COLor:MAXimum

Sets or queries the maximum value of the color axis in the DPX, Spectrum measurement.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax

```
[SENSe]:DPX:COLor:MAXimum <value>
[SENSe]:DPX:COLor:MAXimum?
```

Arguments `<value> ::= <NRf>` specifies the maximum value of the color axis.
Range: The minimum value to 100%.

The minimum value is set using the [SENSe]:DPX:COLor:MINimum command.

Examples `SENSE:DPX:COLOR:MAXIMUM 90` sets the maximum value of the color axis to 90%.

[SENSe]:DPX:COLor:MINimum

Sets or queries the minimum value of the color axis in the DPX, Spectrum measurement.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax

```
[SENSe]:DPX:COLor:MINimum <value>
[SENSe]:DPX:COLor:MINimum?
```

Arguments

<value> :: <NRf> specifies the minimum value of the color axis. Range: 0% to the maximum value.

The maximum value is set using the [SENSe]:DPX:COLor:MAXimum command.

Examples

SENSE:DPX:COLOR:MINIMUM 10 sets the minimum value of the color axis to 10%.

[SENSe]:DPX:DDENsity:MEASurement:SHOW

Sets the Show measurement value (measurement always visible for DPX Trigger) or queries for the current value.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:DDENsity:MEASurement:SHOW { OFF | ON | 0 | 1 }
[SENSe]:DPX:DDENsity:MEASurement:SHOW?
```

Arguments

OFF or 0 specifies that the Show measurement value is disabled.

ON or 1 specifies that the Show measurement value is disabled.

Examples

SENSe:DPX:DDENSITY:MEASUREMENT:SHOW ON enables the Show measurement value.

[SENSe]:DPX:DGRam:COLor:MAXimum

Sets or queries the maximum value of the color axis in the DPXogram measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:DGRam:COLor:MAXimum <value>
[SENSe]:DPX:DGRam:COLor:MAXimum?
```

Arguments

<value> ::= <NRf> specifies the maximum value of the color axis.

Range: –50 dBm to –169.00 dBm.

The minimum value is set using the [SENSe]:DPX:DGRam:COLor:MINimum command.

Examples

`[SENSE]:DPX:DGRAM:COLOR:MAXIMUM 0` sets the maximum value of the color axis to 0.00 dBm.

[SENSe]:DPX:DGRam:COLor:MINimum

Sets or queries the minimum value of the color axis in the DPXogram measurement.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:DGRam:COLor:MINimum <value>
[SENSe]:DPX:DGRam:COLor:MINimum?
```

Arguments

<value> ::= <NRf> specifies the maximum value of the color axis.

Range: 49.00 dBm to –170.00 dBm.

The maximum value is set using the [SENSe]:DPX:DGRam:COLor:MAXimum command.

Examples

`[SENSE]:DPX:DGRAM:COLOR:MINIMUM -100` sets the minimum value of the color axis to –100.00 dBm.

[SENSe]:DPX:DGRam:POINts:COUNt

Sets or queries the number of trace points acquired for the DPXogram display.

Conditions

Measurement views: DPX, Spectrum

Group Sense commands

Syntax
```
[SENSe]:DPX:DGRam:POINts:COUNt { P801 | P2401 | P4001 }
[SENSe]:DPX:DGRam:POINts:COUNt?
```

Arguments `P801` sets the number of sample points to 801.

`P2401` sets the number of sample points to 2401.

`P4001` sets the number of sample points to 4001.

Examples `[SENSE]:DPX:DGRAM:POINTS:COUNT P2401` sets the number of trace points to acquire for the DPXogram display.

[SENSe]:DPX:DGRam:TIME:CAPacity? (Query Only)

Returns the time capacity of the DPXogram measurement.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax
```
[SENSe]:DPX:DGRam:TIME:CAPacity?
```

Arguments None

Returns <Time> ::= <String> The time capacity in hours:minutes:seconds.

Examples `[SENSE]:DPX:DGRAM:TIME:CAPACITY?` might return "`0:25:50`" indicating that the time capacity of the measurement was 0 hours, 25 minutes, and 50 seconds.

[SENSe]:DPX:DGRam:TIME:RESolution

Sets or queries the time resolution of the DPXogram measurement.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax
```
[SENSe]:DPX:DGRam:TIME:RESolution <value>
[SENSe]:DPX:DGRam:TIME:RESolution?
```

Arguments <value> ::= <NRf> specifies the time resolution of the DPXogram measurement.

Examples `[SENSE]:DPX:DGRAM:TIME:RESOLUTION 25.0 ms` sets the time resolution to 25 ms.

[SENSe]:DPX:DGRam:TIME:RESolution:AUTO

Determines whether to set the time resolution automatically or manually in the DPXogram measurement.

When Auto is enabled, the time resolution value is based on the Time/div setting. See the DISPlay:DPX:DGRam:TIME[:SCALe]:PDIVision command.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax
```
[SENSe]:DPX:DGRam:TIME:RESolution:AUTO { OFF | ON | 0 | 1 }
[SENSe]:DPX:DGRam:TIME:RESolution:AUTO?
```

Arguments `OFF` or `0` disables the auto time resolution.

`ON` or `1` enables the auto time resolution.

Examples `[SENSE]:DPX:DGRAM:TIME:RESOLUTION:AUTO ON` sets the timer resolution automatically.

[SENSe]:DPX:DGRam:WATerfall:DIRection

Sets or queries the direction of the Waterfall (DPXogram) display.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax

```
[SENSe]:DPX:DGRam:WATerfall:DIRection { NW | NE }
[SENSe]:DPX:DGRam:WATerfall:DIRection?
```

Arguments

`NW` sets the display direction to northwest.

`NE` sets the display direction to northeast.

Examples

`[SENSE]:DPX:DGRAM:WATERFALL:DIRECTION NW` sets the waterfall display direction to northwest.

[SENSe]:DPX:DGRam:WATerfall:ENABle

Sets or queries the Waterfall (DPXogram) display.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:DGRam:WATerfall:ENABle { OFF | ON | 0 | 1 }
[SENSe]:DPX:DGRam:WATerfall:ENABle?
```

Arguments

`OFF` or `0` disables the Waterfall display.

`ON` or `1` enables the Waterfall display.

Examples

`[SENSE]:DPX:DGRAM:WATERFALL:ENABLE ON` enables the 3-D Waterfall display.

[SENSe]:DPX:DGRam:Y[:SCALe]:AUTO (No Query Form)

Rescales the height axis automatically to fit the waveform to the screen in the DPXogram display.

Conditions

Measurement views: DPX, Spectrum

Group

Sense commands

Syntax

```
[SENSe]:DPX:DGRam:Y[:SCALe]:AUTO
```

Arguments None

Examples `[SENSE]:DPX:DGRAM:Y[:SCALE]:AUTO` rescales the height scale automatically to fit the waveform to the DPXogram display.

[SENSe]:DPX:DGRam:Y[:SCALe]:RESet (No Query Form)

Resets the height scale of the DPXogram display to the default values:

Height position = 0 dB and Height scale = 100 dB.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax `[SENSe]:DPX:DGRam:Y[:SCALe]:RESet`

Arguments None

Examples `[SENSE]:DPX:DGRAM:Y[:SCALE]:RESET` Resets the height scale of the DPXogram display.

[SENSe]:DPX:DWELl

Sets or queries the value of the Dwell time for the DPX, Spectrum measurement.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax
```
[SENSe]:DPX:DWELl <value>
[SENSe]:DPX:DWELl?
```

Arguments `<value> ::  <NRf>`specifies the time the DPX sweep remains in a frequency segment collecting data and updating the bitmap and traces before moving on to the next higher frequency segment.
Range: 50 ms to 100 s per frequency segment.

Examples `SENSE:DPX:DWELl 100ms` sets the dwell value for sweeps to 100 ms.

[SENSe]:DPX:DWELl:AUTO

Sets the value of the Dwell time automatically or queries for the current value.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax

```
[SENSe]:DPX:DWELl:AUTO { OFF | ON | 0 | 1 }
[SENSe]:DPX:DWELl:AUTO?
```

Related Commands [SENSe]:DPX:DWELl

Arguments `OFF` or `0` specifies that the dwell is set manually.

`ON` or `1` specifies that the dwell is set automatically.

Examples `SENSE:DPX:DWELl:AUTO ON` sets the dwell value automatically for sweeps.

[SENSe]:DPX:FREQuency:CENTer

Sets or queries the center frequency in the DPX, Spectrum measurement.

NOTE. *The center, start and stop frequencies are set interlocking each other with the following relationships: (start frequency) = (center frequency) - (span)/2 and (stop frequency) = (center frequency) + (span)/2.*

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax

```
[SENSe]:DPX:FREQuency:CENTer <value>
[SENSe]:DPX:FREQuency:CENTer?
```

Related Commands [SENSe]:DPX:FREQuency:STARt, [SENSe]:DPX:FREQuency:STOP

Arguments `<value> ::  <NRf>` specifies the center frequency.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `SENSE:DPX:FREQUENCY:CENTER 7.5GHz` sets the center frequency to 7.5 GHz.

[SENSe]:DPX:FREQuency:SPAN

Sets or queries the frequency span in the DPX, Spectrum measurement.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax
```
[SENSe]:DPX:FREQuency:SPAN <value>
[SENSe]:DPX:FREQuency:SPAN?
```

Arguments `<value> ::  <NRf>` is the frequency span.
Range: 10 Hz to 40 MHz (Standard) / 110 MHz (Option 110)

Examples `SENSE:DPX:FREQUENCY:SPAN 20MHz` sets the span to 20 MHz.

[SENSe]:DPX:FREQuency:SPAN:{BANDwidth|BWIDth}[:RESolution]:RATio

Determines whether to set the resolution bandwidth (RBW) automatically or manually in the DPX, Spectrum measurement.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax
```
[SENSe]:DPX:FREQuency:SPAN:{BANDwidth|BWIDth}[:RESolution]:
RATio <value>
```

```
[SENSe]:DPX:FREQuency:SPAN:{BANDwidth|BWIDth}[:RESolution]:
RATio?
```

Related Commands [SENSe]:DPX:{BANDwidth|BWIDth}[:RESolution]:AUTO

Arguments <value> :: <NRf> is used to calculate the RBW.

Examples SENSE:DPX:FREQuency:SPAN:BANDWIDTH:RATIO 100 the ratio of 100 is used to set the resolution bandwidth when [SENSe]:DPX:{BANDwidth|BWIDth}[:RESolution]:AUTO is set to ON.

[SENSe]:DPX:FREQuency:STARt

Sets or queries the measurement start frequency (left edge on the graph) in the DPX, Spectrum measurement.

The center, start and stop frequencies are set interlocking each other. Refer to the [SENSe]:DPX:FREQuency:CENTer command.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax

```
[SENSe]:DPX:FREQuency:STARt <value>
[SENSe]:DPX:FREQuency:STARt?
```

Related Commands [SENSe]:DPX:FREQuency:STOP

Arguments <value> :: <NRf> is the measurement start frequency.
Range: (center frequency) ± (span)/2.

Examples SENSE:DPX:FREQUENCY:START 6.95GHz sets the start frequency to 6.95 GHz.

[SENSe]:DPX:FREQuency:STEP

Sets or queries the frequency step size (the amount per press by which the up or down key changes the setting value). Programming a specified step size sets [SENSe]:DPX:FREQuency:STEP:AUTO OFF.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax
```
[SENSe]:DPX:FREQuency:STEP <value>
[SENSe]:DPX:FREQuency:STEP?
```

Related Commands [SENSe]:DPX:FREQuency:STEP:AUTO

Arguments `<value> ::  <NRf>` specifies the frequency step size.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `SENSE:DPX:FREQUENCY:STEP 1.5kHz` sets the step size to 1.5 kHz.

[SENSe]:DPX:FREQuency:STEP:AUTO

Determines whether to set the frequency step size automatically or manually.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax
```
[SENSe]:DPX:FREQuency:STEP:AUTO { OFF | ON | 0 | 1 }
[SENSe]:DPX:FREQuency:STEP:AUTO?
```

Arguments `OFF` or `0` specifies that the frequency step size is set manually using the [SENSe]:DPX:FREQuency:STEP command.

`ON` or `1` specifies that the frequency step size is set automatically.

Examples `SENSE:DPX:BANDWIDTH:AUTO ON` sets the frequency step size automatically.

[SENSe]:DPX:FREQuency:STOP

Sets or queries the measurement stop frequency (right edge of the graph) in the DPX, Spectrum measurement.

The center, start and stop frequencies are set interlocking each other. Refer to the [SENSe]:DPX:FREQuency:CENTer command.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax

```
[SENSe]:DPX:FREQuency:STOP <value>
[SENSe]:DPX:FREQuency:STOP?
```

Related Commands [SENSe]:DPX:FREQuency:STARt

Arguments `<value> ::` `<NRf>` is the measurement stop frequency.
Range: (center frequency) ± (span)/2.

Examples `SENSE:DPX:FREQUENCY:STOP 7.05GHz` sets the stop frequency to 7.05 GHz.

[SENSe]:DPX:PLOT

Sets or queries the plot in the DPX, Spectrum measurement.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax

```
[SENSe]:DPX:PLOT { DGRam | SPLit | SPECtrum | ZSPan |
FREQuency | PHASe }
[SENSe]:DPX:PLOT?
```

Arguments `DGRam` selects the DPXogram display.

`SPLit` selects the DPX Split display.

`SPECtrum` selects the DPX, Spectrum display.

`ZSPan` select the DPX Zero Span display.

FREQuency selects the DPX Frequency vs Time display.

PHASe selects the DPX Phase vs Time display.

Examples SENSE:DPX:PLOT SPECTRUM selects the DPX, Spectrum display.

[SENSe]:DPX:POINts:COUNt

Sets the number of trace points acquired for the DPX, Spectrum display.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax
```
[SENSe]:DPX:POINts:COUNt { P801 | P2401 | P4001 | P10401 }
[SENSe]:DPX:POINts:COUNt?
```

Arguments P801 sets the number of sample points to 801.

P2401 sets the number of sample points to 2401.

P4001 sets the number of sample points to 4001.

P8001 sets the number of sample points to 8001.

P10401 sets the number of sample points to 10401.

Examples SENSE:DPX:POINTS:COUNT P2401 sets the number of trace points to acquire for the DPX, Spectrum display.

[SENSe]:DPX:TDM:{BANDwidth|BWIDth}

Sets or queries the value of the measurement bandwidth in the DPX measurement.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax
```
[SENSe]:DPX:TDM:{BANDwidth|BWIDth} <value>
[SENSe]:DPX:TDM:{BANDwidth|BWIDth}?
```

Arguments <value>::<NRf> specifies the measurement bandwidth.

Range: 100 Hz to Max AcqBW.

Examples SENSE:DPX:TDM:BANDWIDTH 200 sets the measurement bandwidth to 200 Hz.

[SENSe]:DPX:TDM:{BANDwidth|BWIDth}:ACTual? (Query Only)

Queries the actual bandwidth in the DPX measurement.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax [SENSe]:DPX:TDM:{BANDwidth|BWIDth}:ACTual?

Arguments None

Returns <NRf> The actual RBW in Hz.

Examples SENSE:DPX:TDM:BANDWIDTH:ACTUAL? might return 10E+3, indicating that the actual bandwidth is 10 kHz

[SENSe]:DPX:TDM:FREQuency:CENTer

Sets or queries the center frequency in the DPX, Spectrum measurement.

NOTE. *The center, start and stop frequencies are set interlocking each other with the following relationships: (start frequency) = (center frequency) - (span)/2 and (stop frequency) = (center frequency) + (span)/2.*

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax
```
[SENSe]:DPX:TDM:FREQuency:CENTer <value>
[SENSe]:DPX:TDM:FREQuency:CENTer?
```

Arguments `<value>::<NRf>` specifies the center frequency.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `SENSE:DPX:TDM:FREQUENCY:CENTER 7.5GHz` sets the center frequency to 7.5 GHz.

[SENSe]:DPX:TDM:FREQuency:TXBer:CLEar:RESults (No Query Form)

Clears the measurement results of the active Tx BER measurement. Sending this command does not halt the Tx BER measurement.

Conditions Measurement views: DPX frequency

Group Sense commands

Syntax `[SENSe]:DPX:TDM:FREQuency:TXBer:CLEar:RESults`

Arguments none

Examples `[SENSE]:DPX:TDM:FREQUENCY:TXBER:CLEAR:RESULTS`

clears the results of the active TxBER measurement, but does not halt the measurement.

[SENSe]:DPX:TDM:FREQuency:TXBer:ENABle

Sets or queries the Tx BER measurement.

Conditions Measurement views: DPX frequency

Group Sense commands

Syntax `[SENSe]:DPX:TDM:FREQuency:TXBer:ENABle { OFF | ON | 0 | 1 }`

Arguments `OFF` or `0` halts the Tx BER measurement.

`ON` or `1` starts the Tx BER measurement.

Returns

Examples `[SENSE]:DPX:TDM:FREQUENCY:TXBER:ENABLE`

[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern

Sets or queries the Tx BER data pattern.

Conditions Measurement views: DPX frequency

Group Sense commands

Syntax
```
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern { PRBS9 | PRBS11 |
PRBS15 | USER }
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern?
```

Arguments `PRBS9` specifies a 9-bit PRBS as the data pattern.

`PRBS11` specifies a 11-bit PRBS as the data pattern.

`PRBS15` specifies a 15-bit PRBS as the data pattern

`USER` specifies a user-defined sequence as the data pattern.

Returns `<string>` which is the selected data pattern. Returned strings are the same as the arguments.

Examples `[SENSE]:DPX:TDM:FREQUENCY:TXBER:PATTERN PRBS9` sets the data pattern to PRBS9.

[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:INVert

Sets or queries the data pattern Invert setting.

Conditions Measurement views: DPX frequency

Group Sense commands

Syntax
```
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:INVert { 0 | 1 | OFF | ON }
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:INVert?
```

Arguments OFF or 0 sets the data pattern to normal.

ON or 1 inverts the data pattern.

Returns <value> ::= <NR1> where:

0 indicates the data pattern is normal.

1 indicates the data pattern is inverted.

Examples [SENSE]:DPX:TDM:FREQUENCY:TXBER:PATTERN:INVERT 1 inverts the data pattern.

[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:REPeat

Sets or queries the test duration for a User Pattern. A User Pattern test duration can be one time through or it can repeat.

Conditions Measurement views: DPX frequency

Group Sense commands

Syntax
```
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:REPeat { 0 | 1 | OFF | ON }
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:REPeat?
```

Arguments OFF or 0 sets the data pattern to normal.

ON or 1 inverts the data pattern.

Returns <value> ::= <NR1> where:

0 indicates repeat is off.

1 indicates repeat is on.

Examples `[SENSE]:DPX:TDM:FREQUENCY:TXBER:PATTERN:REPEAT 1` causes the User pattern to repeat.

[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:REVerse

Sets or queries whether the data pattern is in normal order or reversed. This setting applies only to PRBS patterns.

Conditions Measurement views: DPX frequency

Group Sense commands

Syntax
```
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:REVerse { 0 | 1 | OFF | ON }
[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:REVerse?
```

Arguments `OFF` or `0` sets the data pattern to normal order.

`ON` or `1` sets the data pattern to reversed order.

Returns <value> ::= <NR1> where:

`0` indicates the data pattern is in normal order.

`1` indicates the data pattern is in reversed order.

Examples `[SENSE]:DPX:TDM:FREQUENCY:TXBER:PATTERN:REVERSE?` might return 1, indicating the data pattern is reversed.

[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:SOURce? (Query Only)

Returns the path and file name of the user-defined pattern file.

Conditions Measurement views: DPX frequency

Group Sense commands

Syntax `[SENSe]:DPX:TDM:FREQuency:TXBer:PATTern:SOURce?`

Returns `<path> :: <string>` which is the path to the defined User data pattern.

Examples `[SENSE]:DPX:TDM:FREQUENCY:TXBER:PATTERN:SOURCE?` might return `"C:\RSA5100B Files\Example Files\SampleUserPattern.csv"`.

[SENSe]:DPX:TDM:RBW

Sets or queries the value of the RBW filter.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax
```
[SENSe]:DPX:TDM:RBW <value>
[SENSe]:DPX:TDM:RBW?
```

Arguments `<value>::<NRf>` specifies the RBW filter value.

Range: 1 Hz to 60 MHz

Examples `SENSE:DPX:TDM:RBW 2.5` sets the RBW filter value to 2.5 Hz.

[SENSe]:DPX:TDM:RBW:STATe

Determines whether to set the resolution bandwidth (RBW) filter automatically or manually in the DPX measurement.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax
```
[SENSe]:DPX:TDM:RBW:STATe { OFF | ON | 0 | 1 }
[SENSe]:DPX:TDM:RBW:STATe?
```

Arguments `OFF` or `0` specifies that the RBW is set automatically.

`ON` or `1` specifies that the RBW is set manually.

Examples `SENSE:DPX:TDM:RBW:STATE ON` sets the RBW manually.

[SENSe]:DPX:TDM:SWEep:TIME

Sets or queries the sweep time when the plot is set to ZSPan, FREQuency, or PHASe.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax
```
[SENSe]:DPX:TDM:SWEep:TIME <value>
[SENSe]:DPX:TDM:SWEep:TIME?
```

Arguments `<value>::<NRf>` specifies the sweep time.

Range: 100 ηs to 2000 s.

Examples `SENSE:DPX:TDM:SWEEP:TIME 200` sets the sweep time to 200 s.

[SENSe]:DPX:TDM:TMOTion

Sets or queries the trace motion when the plot is set to ZSPan, FREQuency, or PHASe.

Conditions Measurement views: DPX, Spectrum

Group Sense commands

Syntax
```
[SENSe]:DPX:TDM:TMOTion { NONE | ROLL | NORMal }
[SENSe]:DPX:TDM:TMOTion?
```

Arguments `NONE` turns scrolling off.

`ROLL` shifts the previous data points to the left and the latest data point to the right edge of the display.

`NORMAL` shifts the latest data point on the display from left to right.

Examples `[SENSE]:DPX:TDM:TMOTION NORMAL` sets the scroll direction to normal.

[SENSe]:{FM|PM}:BURSt:THReshold

Sets or queries the threshold level above which the input signal is determined to be a burst in the FM/PM measurement. The burst detected first is used for the measurement.

Conditions Measurement views: FM, PM

Group Sense commands

Syntax
```
[SENSe]:{FM|PM}:BURSt:THReshold <value>
[SENSe]:{FM|PM}:BURSt:THReshold?
```

Arguments `<value> ::` `<NRf>` specifies the threshold level. Range: –100.0 to –10.0 dB.

Examples `SENSE:FM:BURST:THRESHOLD -10` sets the threshold level to –10 dB in the FM measurement.

[SENSe]:{FM|PM}:FREQuency:OFFSet

Sets or queries the carrier frequency offset in the FM/PM measurement.

Conditions Measurement views: FM, PM

Group Sense commands

Syntax
```
[SENSe]:{FM|PM}:FREQuency:OFFSet <value>
[SENSe]:{FM|PM}:FREQuency:OFFSet?
```

Arguments `<value> ::` `<NRf>` specifies the frequency offset from the center frequency. Range: -(MeasBW*1.1)/2 to +(MeasBW*1.1)/2 where MeasBW is set by the command [SENSe]:{AM|FM|PM}:{BANDwidth|BWIDth}:MEASurement.

Examples `SENSE:FM:FREQUENCY:OFFSET 10MHZ` sets the carrier frequency offset to 10 MHz.

[SENSe]:{FM|PM}:FREQuency:OFFSet:MARKer (No Query Form)

Sets the frequency offset from the selected marker location in the FM measurement.

Sets the frequency offset from the selected delta marker location in the PM measurement.

Conditions Measurement views: FM, PM

Group Sense commands

Syntax `[SENSe]:{FM|PM}:FREQuency:OFFSet:MARKer`

Arguments None.

Examples `SENSE:FM:FREQUENCY:OFFSET:MARKER` sets frequency offset from the selected Marker position.

[SENSe]:{FM|PM}:FREQuency:SEARch:AUTO

Determines whether to detect the carrier frequency automatically or manually in the FM/PM measurements.

Conditions Measurement views: FM, PM

Group Sense commands

Syntax
```
[SENSe]:{FM|PM}:FREQuency:SEARch:AUTO { OFF | ON | 0 | 1 }
[SENSe]:{FM|PM}:FREQuency:SEARch:AUTO?
```

Arguments `OFF` or `0` specifies that the carrier frequency is set manually. Use the command[SENSe]:{FM|PM}:FREQuency:OFFSet to set the carrier offset frequency.

`ON` or `1` specifies that the carrier frequency is detected automatically.

Examples `SENSE:FM:FREQUENCY:SEARCH:AUTO ON` specifies that the carrier frequency is detected automatically.

[SENSe]:{FSETtling|PSETtling}:{BANDwidth|BWIDTH}

Sets or queries the measurement bandwidth for the Settling Time measurements.

Conditions Measurement views: Frequency and Phase Settling Time

Group Sense commands

Syntax

```
[SENSe]:{FSETtling|PSETtling}:{BANDwidth|BWIDTH} <value>
[SENSe]:{FSETtling|PSETtling}:{BANDwidth|BWIDTH}?
```

Related Commands FETCh:{AM|FM|PM}?

READ:AM:RESult?

Arguments `<value> ::` `<NRf>` specifies the bandwidth for the specified settling time measurement.

Examples `SENSE:FSETTLING:BANDwidth 1e6` sets the Frequency Settling time measurement bandwidth to 1 MHz.

[SENSe]:{FSETtling|PSETtling}:{BANDwidth|BWIDTH}:ACTual? (Query Only)

Queries the actual measurement bandwidth in the Settling Time measurements.

Conditions Measurement views: Frequency and Phase Settling Time

Group Sense commands

Syntax

```
[SENSe]:{FSETtling|PSETtling}:{BANDwidth|BWIDTH}:ACTual?
```

Arguments None

Returns `<NRf>` The actual measurement bandwidth in Hz.

Examples `SENSE:FSETTLING:BANDWIDTH:ACTUAL?` might return `40.000E+6`, indicating that the actual RBW is 40.00 MHz.

[SENSe]:{FSETtling|PSETtling}:FREQuency:CENTer

Sets or queries the measurement frequency.

Conditions Measurement views: Frequency and Phase Settling Time

Group Sense commands

Syntax

```
[SENSe]:{FSETtling|PSETtling}:FREQuency:CENTer <value>
[SENSe]:{FSETtling|PSETtling}:FREQuency:CENTer?
```

Arguments <value> :: <NRf> specifies the measurement frequency.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples SENSE:PSETTLING:FREQUENCY:CENTER 7.5GHz sets the Phase Settling Time measurement frequency to 7.5 GHz.

[SENSe]:{FSETtling|PSETtling}:FREQuency:OFFSet

Sets or queries the offset frequency.

Conditions Measurement views: Frequency and Phase Settling Time

Group Sense commands

Syntax

```
[SENSe]:{FSETtling|PSETtling}:FREQuency:OFFSet <value>
[SENSe]:{FSETtling|PSETtling}:FREQuency:OFFSet?
```

Arguments <value> :: <NRf> specifies the offset frequency.
Range: ± (0.5 × Measurement bandwidth)

Default: 0 Hz.

Examples `SENSE:PSETTLING:FREQUENCY:OFFSET 20E6` sets the Phase Settling Time offset frequency to 20 MHz.

[SENSe]:{FSETtling|PSETtling}:LENGth

Sets or queries the measurement length.

Conditions Measurement views: Frequency and Phase Settling Time

Group Sense commands

Syntax
```
[SENSe]:{FSETtling|PSETtling}:LENGth <value>
[SENSe]:{FSETtling|PSETtling}:LENGth?
```

Arguments `<value> ::  <NRf>` specifies the measurement length in seconds.
Range: 10 ns to [(acquisition length) - 400 ns].

If [(analysis start) + (analysis length)] > [(acquisition length) - 400 ns], the actual measurement length is reduced to [(acquisition length) - 200 ns].

Default: 1 ms.

Examples `SENSE:PSETTLING:LENGTH 25.625us` sets the measurement length to 25.625 µs.

[SENSe]:{FSETtling|PSETtling}:LENGth:ACTual? (Query Only)

Queries the actual measurement length for the Settling Time measurements.

Conditions Measurement views: Frequency and Phase Settling Time

Group Sense commands

Syntax
```
[SENSe]:{FSETtling|PSETtling}:LENGth:ACTual?
```

Arguments None

Returns `<NRf>` The actual measurement length in Hz.

Examples `SENSE:FSETTLING:LENGTH:ACTUAL?` might return `25.625E-6`, indicating that the actual measurement length is 25.625 µs.

[SENSe]:{FSETtling|PSETtling}:MASK:LIMit

Sets or queries the frequency or phase limit for the specified region.

Conditions Measurement views: Frequency and Phase Settling Time

Group Sense commands

Syntax

```
[SENSe]:{FSETtling|PSETtling}:MASK:LIMit <x> <value>
[SENSe]:{FSETtling|PSETtling}:MASK:LIMit <x>?
```

Arguments `<x>::= { 1 | 2 | 3}` specifies the mask region.

`<value> :: <NRf>` specifies the frequency or phase limit for the region.

For Frequency Settling Time:

- Range: 1 Hz to 1 GHz
- Default: 500 kHz

For Phase Settling Time:

- Range: 0.1 degree to 180 degrees
- Default: 1 degree

Examples `SENSE:FSETTLING:MASK:LIMIT2 7.5E5` sets the limit for region 2 to 750 kHz.

[SENSe]:{FSETtling|PSETtling}:MASK:STATe

Sets or queries mask test state.

Conditions Measurement views: Frequency and Phase Settling Time

Group Sense commands

Syntax

```
[SENSe]:{FSETtling|PSETtling}:MASK:STATe { OFF | ON | 0 |
1 }
```

Arguments `OFF` or `0` enables Mask test.

`ON` or `1` disables Mask test.

Examples `SENSE:FSETTLING:MASK:STATE ON` enables mask test.

[SENSe]:{FSETtling|PSETtling}:MASK:TIME:REFerence

Sets or queries the mask time reference.

Conditions Measurement views: Frequency and Phase Settling Time

Group Sense commands

Syntax
```
[SENSe]:{FSETtling|PSETtling}:MASK:TIME:REFerence { TRIGger
| TRANsition }
```

Arguments `TRIGger` specifies that the trigger point service as the mask time reference.

`TRANsition` specifies that the point where the signal transitions out of the previous tolerance range serves as the mask time reference.

Default: Trigger

Examples `SENSE:FSETTLING:MASK:TIME:REFerence TRIGGER` specifies that the trigger point serve as the mask time reference.

[SENSe]:{FSETtling|PSETtling}:MASK:TIME:STARt

Sets or queries the mask region start time in seconds.

Conditions Measurement views: Frequency and Phase Settling Time

Group Sense commands

Syntax
```
[SENSe]:{FSETtling|PSETtling}:MASK:TIME:STARt <x> <value>
[SENSe]:{FSETtling|PSETtling}:MASK:TIME:STARt <x>?
```

Arguments <x>::= { 1 | 2 | 3} specifies the mask region.

<value> :: <NRf> specifies the start time of the mask region in seconds.

Examples SENSE:FSETTLING:MASK:TIME:START3? might return 1.0000E-3 as the start time for mask region 3.

[SENSe]:{FSETtling|PSETtling}:MASK:TIME:STOP

Sets or queries the mask region stop time in seconds.

Conditions Measurement views: Frequency and Phase Settling Time

Group Sense commands

Syntax
```
[SENSe]:{FSETtling|PSETtling}:MASK:TIME:STOP <value>
[SENSe]:{FSETtling|PSETtling}:MASK:TIME:STOP <x>?
```

Arguments <value> :: <NRf> specifies the stop time of the mask in seconds.

Examples SENSE:FSETTLING:MASK:TIME:STOP? might return 6.244120000E-3 as the mask stop time.

[SENSe]:{FSETtling|PSETtling}:MAXTracepoints

Sets or queries the maximum trace points in the Settling Time measurements.

Conditions Measurement views: Frequency and Phase Settling

Group Sense commands

Syntax
```
[SENSe]:{FSETtling|PSETtling}:MAXTracepoints { ONEK | TENK
| HUNDredk | NEVerdecimate }
[SENSe]:{FSETtling|PSETtling}:MAXTracepoints?
```

Arguments ONEK sets the maximum trace points to 1 k.

TENK sets the maximum trace points to 10 k.

HUNDredk sets the maximum trace points to 100 k.

NEVerdecimate never decimates the trace points.

Examples SENSE:PSETTLING:MAXTRACEPOINTS TENK sets the maximum trace points to 10 k.

[SENSe]:{FSETtling|PSETtling}:SDURation:MINimum

Sets or queries the minimum settled duration in seconds. This duration time determines the result of :FETCh: {FSETtling | PSETtling}:SLMSd[:PASS]?.

Conditions Measurement views: Frequency and Phase Settling Time

Group Sense commands

Syntax
```
[SENSe]:{FSETtling|PSETtling}:SDURation:MINimum <value>
[SENSe]:{FSETtling|PSETtling}:SDURation:MINimum?
```

Arguments <value> :: <NRf> specifies the minimum settled duration in seconds.

Examples SENSE:FSETTLING:SDURATION:MINIMUM 17.73E-3 sets the minimum settled duration to 17.73 ms.

[SENSe]:{FSETtling|PSETtling}:TARGet:REFerence

Sets or queries the target reference frequency.

Conditions Measurement views: Frequency and Phase Settling Time

Group Sense commands

Syntax
```
[SENSe]:{FSETtling|PSETtling}:TARGet:REFerence { AUTO |
MFReq }
[SENSe]:{FSETtling|PSETtling}:TARGet:REFerence?
```

Arguments AUTO causes the instrument to determine the target reference value by averaging at least the last 256 samples at the end of the measurement period.

MFReq specifies that the Measurement Frequency value is used as the target reference frequency.

Default: AUTO

Examples SENSE:FSETTLING:TARGET:REFERENCE MFREQ specifies that the target reference frequency is the measurement frequency.

[SENSe]:{FSETtling|PSETtling}:TOLerance

Sets or queries the frequency or phase range into which the signal must settle.

Conditions Measurement views: Frequency and Phase Settling Time

Group Sense commands

Syntax

```
[SENSe]:{FSETtling|PSETtling}:TOLerance <value>
[SENSe]:{FSETtling|PSETtling}:TOLerance?
```

Arguments <value> :: <NRf> specifies the frequency or phase range.

For Frequency Settling Time:

- Range: 1 Hz to 1 GHz
- Default: 500 kHz

For Phase Settling Time:

- Range: 0.1 degree to 180 degrees
- Default: 1 degree

Examples SENSE:FSETTLING:TOLERANCE 2.5E5 sets the tolerance to 250 kHz.

[SENSe]:FVTime:CLEar:RESults (No Query Form)

Restarts multi-trace functions (Average and Max/Min Hold).

Conditions Measurement views: Frequency versus Time

Group Sense commands

Syntax `[SENSe]:FVTime:CLEar:RESults`

Arguments None

Examples `SENSE:FVTIME:CLEAR:RESULTS` restarts multi-trace functions.

[SENSe]:FVTime:FREQuency:CENTer

Sets or queries the center frequency in the Frequency versus Time measurement.

NOTE. *The center, start and stop frequencies are set interlocking each other with the following relationships: (start frequency) = (center frequency) - (span)/2 and (stop frequency) = (center frequency) + (span)/2.*

Conditions Measurement views: Frequency versus Time

Group Sense commands

Syntax
```
[SENSe]:FVTime:FREQuency:CENTer <value>
[SENSe]:FVTime:FREQuency:CENTer?
```

Related Commands [SENSe]:FVTime:FREQuency:STARt, [SENSe]:FVTime:FREQuency:STOP

Arguments `<value> ::  <NRf>` specifies the center frequency.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `SENSE:FVTIME:FREQUENCY:CENTER 7.5GHz` sets the center frequency to 7.5 GHz.

[SENSe]:FVTime:FREQuency:SPAN

Sets or queries the frequency span in the Frequency versus Time measurement.

Conditions Measurement views: Frequency versus Time

Group Sense commands

Syntax
```
[SENSe]:FVTime:FREQuency:SPAN <value>
[SENSe]:FVTime:FREQuency:SPAN?
```

Arguments <value> :: <NRf> is the frequency span.
Range: 10 Hz to 40 MHz (Standard) / 110 MHz (Option 110)

Examples SENSE:FVTIME:FREQUENCY:SPAN 20MHz sets the span to 20 MHz.

[SENSe]:FVTime:FREQuency:STARt

Sets or queries the measurement start frequency (left edge on the graph) in the Frequency versus Time measurement.

The center, start and stop frequencies are set interlocking each other. Refer to the [SENSe]:FVTime:FREQuency:CENTer command.

Conditions Measurement views: Frequency versus Time

Group Sense commands

Syntax
```
[SENSe]:FVTime:FREQuency:STARt <value>
[SENSe]:FVTime:FREQuency:STARt?
```

Related Commands [SENSe]:FVTime:FREQuency:STOP

Arguments <value> :: <NRf> is the measurement start frequency.
Range: (center frequency) ± (span)/2.

Examples SENSE:FVTIME:FREQUENCY:START 6.95GHz sets the start frequency to 6.95 GHz.

[SENSe]:FVTime:FREQuency:STEP

Sets or queries the frequency step size (the amount per press by which the up or down key changes the setting value). Programming a specified step size sets [SENSe]:FVTime:FREQuency:STEP:AUTO OFF.

Conditions Measurement views: Frequency versus Time

Group Sense commands

Syntax

```
[SENSe]:FVTime:FREQuency:STEP <value>
[SENSe]:FVTime:FREQuency:STEP?
```

Related Commands [SENSe]:FVTime:FREQuency:STEP:AUTO

Arguments `<value> ::  <NRf>` specifies the frequency step size.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `SENSE:FVTIME:FREQUENCY:STEP 1.5kHz` sets the step size to 1.5 kHz.

[SENSe]:FVTime:FREQuency:STEP:AUTO

Determines whether to set the frequency step size automatically or manually.

Conditions Measurement views: Frequency versus Time

Group Sense commands

Syntax

```
[SENSe]:FVTime:FREQuency:STEP:AUTO { OFF | ON | 0 | 1 }
[SENSe]:FVTime:FREQuency:STEP:AUTO?
```

Arguments `OFF` or `0` specifies that the frequency step size is set manually using the [SENSe]:FVTime:FREQuency:STEP command.

ON or 1 specifies that the frequency step size is set automatically.

Examples SENSE:FVTIME:BANDWIDTH:AUTO ON sets the frequency step size automatically.

[SENSe]:FVTime:FREQuency:STOP

Sets or queries the measurement stop frequency (right edge of the graph) in the Frequency versus Time measurement.

The center, start and stop frequencies are set interlocking each other. Refer to the [SENSe]:FVTime:FREQuency:CENTer command.

Conditions Measurement views: Frequency versus Time

Group Sense commands

Syntax
```
[SENSe]:FVTime:FREQuency:STOP <value>
[SENSe]:FVTime:FREQuency:STOP?
```

Related Commands [SENSe]:FVTime:FREQuency:STARt

Arguments <value> :: <NRf> is the measurement stop frequency.
Range: (center frequency) ± (span)/2.

Examples SENSE:FVTIME:FREQUENCY:STOP 7.05GHz sets the stop frequency to 7.05 GHz.

[SENSe]:FVTime:MAXTracepoints

Sets or queries the maximum trace points in the Frequency versus Time measurement.

Conditions Measurement views: Frequency versus Time

Group Sense commands

Syntax
```
[SENSe]:FVTime:MAXTracepoints { ONEK | TENK | HUNDredk |
NEVerdecimate }
```

[SENSe]:FVTime:MAXTracepoints?

Arguments ONEK sets the maximum trace points to 1 k.

TENK sets the maximum trace points to 10 k.

HUNDredk sets the maximum trace points to 100 k.

NEVerdecimate never decimates the trace points.

Examples SENSE:FVTIME:MAXTRACEPOINTS TENK sets the maximum trace points to 10 k.

[SENSe]:IQVTime:CLEar:RESults (No Query Form)

Restarts multi-trace functions (Average and Max/Min Hold).

Conditions Measurement views: RF I&Q versus Time

Group Sense commands

Syntax [SENSe]:IQVTime:CLEar:RESults

Arguments None

Examples SENSE:IQVTIME:CLEAR:RESULTS restarts multi-trace functions.

[SENSe]:IQVTime:FREQuency:CENTer

Sets or queries the center frequency in the RF I&Q versus Time measurement.

NOTE. *The center, start and stop frequencies are set interlocking each other with the following relationships: (start frequency) = (center frequency) - (span)/2 and (stop frequency) = (center frequency) + (span)/2.*

Conditions Measurement views: RF I&Q versus Time

Group Sense commands

Syntax

```
[SENSe]:IQVTime:FREQuency:CENTer <value>
[SENSe]:IQVTime:FREQuency:CENTer?
```

Related Commands [SENSe]:IQVTime:FREQuency:STARt, [SENSe]:IQVTime:FREQuency:STOP

Arguments `<value> ::` `<NRf>` specifies the center frequency.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `SENSE:IQVTIME:FREQUENCY:CENTER 7.5GHz` sets the center frequency to 7.5 GHz.

[SENSe]:IQVTime:FREQuency:SPAN

Sets or queries the frequency span in the RF I&Q versus Time measurement.

Conditions Measurement views: RF I&Q versus Time

Group Sense commands

Syntax

```
[SENSe]:IQVTime:FREQuency:SPAN <value>
[SENSe]:IQVTime:FREQuency:SPAN?
```

Arguments `<value> ::` `<NRf>` is the frequency span.
Range: 10 Hz to 40 MHz (Standard) / 110 MHz (Option 110)

Examples `SENSE:IQVTIME:FREQUENCY:SPAN 20MHz` sets the span to 20 MHz.

[SENSe]:IQVTime:FREQuency:STARt

Sets or queries the measurement start frequency (left edge on the graph) in the RF I&Q versus Time measurement.

The center, start and stop frequencies are set interlocking each other. Refer to the [SENSe]:IQVTime:FREQuency:CENTer command.

Conditions Measurement views: RF I&Q versus Time

Group Sense commands

Syntax
```
[SENSe]:IQVTime:FREQuency:STARt <value>
[SENSe]:IQVTime:FREQuency:STARt?
```

Related Commands [SENSe]:IQVTime:FREQuency:STOP

Arguments `<value> ::` `<NRf>` is the measurement start frequency.
Range: (center frequency) ± (span)/2.

Examples `SENSE:IQVTIME:FREQUENCY:START 6.95GHz` sets the start frequency to 6.95 GHz.

[SENSe]:IQVTime:FREQuency:STEP

Sets or queries the frequency step size (the amount per press by which the up or down key changes the setting value). Programming a specified step size sets [SENSe]:IQVTime:FREQuency:STEP:AUTO OFF.

Conditions Measurement views: RF I&Q versus Time

Group Sense commands

Syntax
```
[SENSe]:IQVTime:FREQuency:STEP <value>
[SENSe]:IQVTime:FREQuency:STEP?
```

Related Commands [SENSe]:IQVTime:FREQuency:STEP:AUTO

Arguments `<value> ::` `<NRf>` specifies the frequency step size.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `SENSE:IQVTIME:FREQUENCY:STEP 1.5kHz` sets the step size to 1.5 kHz.

[SENSe]:IQVTime:FREQuency:STEP:AUTO

Determines whether to set the frequency step size automatically or manually.

Conditions Measurement views: RF I&Q versus Time

Group Sense commands

Syntax

```
[SENSe]:IQVTime:FREQuency:STEP:AUTO { OFF | ON | 0 | 1 }
[SENSe]:IQVTime:FREQuency:STEP:AUTO?
```

Arguments `OFF` or `0` specifies that the frequency step size is set manually using the [SENSe]:IQVTime:FREQuency:STEP command.

`ON` or `1` specifies that the frequency step size is set automatically.

Examples `SENSE:IQVTIME:FREQUENCY:STEP:AUTO ON` sets the frequency step size automatically.

[SENSe]:IQVTime:FREQuency:STOP

Sets or queries the measurement stop frequency (right edge of the graph) in the RF I&Q versus Time measurement.

The center, start and stop frequencies are set interlocking each other. Refer to the [SENSe]:IQVTime:FREQuency:CENTer command.

Conditions Measurement views: RF I&Q versus Time

Group Sense commands

Syntax

```
[SENSe]:IQVTime:FREQuency:STOP <value>
[SENSe]:IQVTime:FREQuency:STOP?
```

Related Commands [SENSe]:IQVTime:FREQuency:STARt

Arguments <value> :: <NRf> is the measurement stop frequency.
Range: (center frequency) ± (span)/2.

Examples SENSE:IQVTIME:FREQUENCY:STOP 7.05GHz sets the stop frequency to 7.05 GHz.

[SENSe]:IQVTime:MAXTracepoints

Sets or queries the maximum trace points in the RF I&Q versus Time measurement.

Conditions Measurement views: RF I&Q versus Time

Group Sense commands

Syntax
```
[SENSe]:IQVTime:MAXTracepoints { ONEK | TENK | HUNDredk |
NEVerdecimate }
[SENSe]:IQVTime:MAXTracepoints?
```

Arguments ONEK sets the maximum trace points to 1 k.

TENK sets the maximum trace points to 10 k.

HUNDredk sets the maximum trace points to 100 k.

NEVerdecimate never decimates the trace points.

Examples SENSE:IQVTIME:MAXTRACEPOINTS TENK sets the maximum trace points to 10 k.

SENSe:LTE:ACLR:BANDwidth:RESolution

Sets or queries the resolution bandwidth (RBW) in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Sense commands

Syntax SENSe:LTE:ACLR:BANDwidth:RESolution

Arguments <value>::<NRf> specifies the resolution bandwidth

Returns <NR3>

Examples `SENSE:LTE:ACLR:BANDWIDTH:RESOLUTION`? might return 100.0000000E+3, indicating the RBW is 100 kHz.

SENSe:LTE:ACLR:BANDwidth:VIDeo

Sets or queries the video bandwidth (VBW) in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Sense commands

Syntax `SENSe:LTE:ACLR:BANDwidth:VIDeo <value>`

Arguments <value>::<NRf> specifies the video bandwidth

Returns <NR3>

Examples `SENSE:LTE:ACLR:BANDWIDTH:VIDEO 2.0000000E+3` sets the VBW to 2 kHz.

SENSe:LTE:ACLR:BANDwidth:VIDeo:STATe

Sets or queries enabling or disabling the video bandwidth (VBW) in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Sense commands

Syntax
```
SENSe:LTE:ACLR:BANDwidth:VIDeo:STATe { 0 | 1 }
SENSe:LTE:ACLR:BANDwidth:VIDeo:STATe?
```

Arguments 0 disables the VBW.

1 enables the VBW.

Returns 0 means that the VBW is disabled.

1 means that the VBW is enabled.

Examples `SENSE:LTE:ACLR:BANDWIDTH:VIDEO:STATE 1` sets the VBW to automatic (enabled) and is decided by the analysis module.

SENSe:LTE:ACLR:CHANnel:IBANdwidth

Sets or queries the reference channel integration bandwidth for the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Sense commands

Syntax
```
SENSe:LTE:ACLR:CHANnel:IBANdwidth <value>
SENSe:LTE:ACLR:CHANnel:IBANdwidth?
```

Arguments <value>::<NRf> specifies the channel integration bandwidth.

Returns <NR3>

Examples `SENSE:LTE:ACLR:CHANNEL:IBANDWIDTH?` might return 18.000000E+6 indicating that the integration bandwidth is 18 MHz.

SENSe:LTE:ACLR:CHANnel:WIDTh

Sets or queries the reference channel width for the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Sense commands

Syntax

```
SENSe:LTE:ACLR:CHANnel:WIDTh <value>
SENSe:LTE:ACLR:CHANnel:WIDTh?
```

Arguments <value>::<NRf> specifies the reference channel width.

Returns <NR3>

Examples `SENSE:LTE:ACLR:CHANNEL:WIDTH 5.00000000E+6` sets the reference channel width to 5 MHz.

SENSe:LTE:ACLR:CLEar:RESults (No Query Form)

Clears the results in the LTE ACLR measurement.

Conditions Measurement view: LTE ACLR

Group Sense commands

Syntax

```
SENSe:LTE:ACLR:CLEar:RESults
```

Arguments None

Examples `SENSE:LTE:ACLR:CLEAR:RESULTS` clears the results of the LTE ACLR measurement.

SENSe:LTE:ACLR:FREQuency

Sets or queries the measurement center frequency in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Sense commands

Syntax

```
SENSe:LTE:ACLR:FREQuency <value>
SENSe:LTE:ACLR:FREQuency?
```

Arguments <value>::<NRf> specifies the center frequency

Returns <NR3>

Examples `SENSE:LTE:ACLR:FREQUENCY` 1.000000000E+9 sets the center frequency to 1.0 GHz.

SENSe:LTE:ACLR:MFILter

Sets or queries the shape of the filter for the LTE ACLR measurement.

Conditions Measurement view: LTE ACLR

Group Sense commands

Syntax
```
SENSe:LTE:ACLR:MFILter { GAUSsian | RECTangular }
SENSe:LTE:ACLR:MFILter?
```

Arguments GAUSsian: sets the filter shape to Gaussian.

RECTangular: sets the filter shape to rectangular.

Returns GAUSSIAN: means the filter shape is Gaussian.

RECTangular: means the filter shape is rectangular.

Examples `SENSE:LTE:ACLR:MFILTER RECT` sets the filter shape to rectangular.

SENSe:LTE:ACLR:MODE:REALtime

Sets or queries the real-time LTE ACLR measurement mode. When set to real-time, the reference channel and all requested adjacent channel regions are acquired together. When real-time if off, the measurement is done in Swept mode (an acquisition for each adjacent band).

Conditions Measurement view: LTE ACLR

Group Sense commands

Syntax

```
SENSe:LTE:ACLR:MODE:REALtime { OFF | ON | 0 | 1 }
SENSe:LTE:ACLR:MODE:REALtime?
```

Arguments

OFF or 0 turns off the automatic update.

ON or 1 turns on automatic updates center frequency.

Returns

OFF or 0 means automatic update is off.

ON or 1 means automatic updates are off.

Examples

`SENSE:LTE:ACLR:MODE:REALTIME?` might return 1 to indicate the real-time checkbox in the LTE ACLR control panel is enabled.

SENSe:LTE:ACLR:NFLoor:ACQuire (No Query Form)

Enables measurement of the noise floor for the LTE ACLR measurement.

Conditions

Measurement view: LTE ACLR

Group

Sense commands

Syntax

```
SENSe:LTE:ACLR:NFLoor:ACQuire
```

Arguments

None

Examples

`SENSE:LTE:ACLR:NFLOOR:ACQUIRE` will enable measurement of the noise floor in the LTE ACLR measurement.

SENSe:LTE:ACLR:NFLoor:CORRected? (Query Only)

Returns whether or not noise correction was applied on the last acquisition in the LTE ACLR measurement.

Conditions

Measurement view: LTE ACLR

Group

Sense commands

Syntax `SENSe:LTE:ACLR:NFLoor:CORRected?`

Returns 1 means that noise correction was applied on the last acquisition.

0 means that noise correction was not applied on the last acquisition.

Examples `SENSE:LTE:ACLR:NFLOOR:CORRECTED?` might return 1, indicating that noise correction was applied on the last acquisition.

SENSe:LTE:ACLR:NFLoor:STATe

Determines whether to enable or disable the correction for noise floor for the LTE ACLR measurement.

Conditions Measurement view: LTE ACLR

Group Sense commands

Syntax
```
SENSe:LTE:ACLR:NFLoor:STATe { OFF | ON | 0 | 1 }
SENSe:LTE:ACLR:NFLoor:STATe?
```

Arguments OFF or 0 disables noise correction in the measurement view.

ON or 1 enables noise correction in the measurement view.

Returns OFF or 0 means that noise correction is disabled in the measurement view.

ON or 1 means that noise correction is enabled in the measurement view.

Examples `SENSE:LTE:ACLR:NFLOOR:STATE OFF` disables noise correction in the measurement view.

SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution

Sets or queries the resolution bandwidth (RBW) value in the specified row in the Offsets & Limits table for the LTE ACLR display.

Conditions Measurement view: LTE ACLR

The parameter <x> represents the row in the Offsets & Limits table.

Group Sense commands

Syntax
```
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution <value>
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution?
```

Arguments <value>::<NRf> specifies the RBW in the specified row.

Returns <NR3>

Examples `SENSE:LTE:ACLR:OFFSET4:BANDWIDTH:RESOLUTION 30.000000000E+3` sets the RBW to 30 kHz for row 4.

SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:COUNt

Sets or queries the count for the resolution bandwidths in the specified row in the Offsets & Limits table for the LTE ACLR display.

Conditions Measurement view: LTE ACLR

The parameter <x> represents the row in the Offsets & Limits table.

Group Sense commands

Syntax
```
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:COUNt <value>
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:COUNt?
```

Arguments <value>::<NR1> specifies a count integer

Returns <NR1>

Examples `SENSE:LTE:ACLR:OFFSET1:BANDWIDTH:RESOLUTION:COUNT 10` sets the count for row 1 to 10.

SENSe:LTE:ACLR:OFFSet<X>:BANDwidth:RESolution:FILTer

Sets or queries the filter bandwidth resolution under Offset & Limit table tab for the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Parameter <x> represents the row in the Offsets & Limits table

Group Sense commands

Syntax
```
SENSe:LTE:ACLR:OFFSet<X>:BANDwidth:RESolution:FILTer <value>
SENSe:LTE:ACLR:OFFSet<X>:BANDwidth:RESolution:FILTer?
```

Arguments <value>::<NRf> specifies the filter bandwidth resolution in the Offset & Limit table.

Returns <NR3>

Examples `SENSE:LTE:ACLR:OFFSET1:BANDWIDTH:RESOLUTION:FILTER?` might return 100.0000000000E+3 indicating the value of filter bandwidth of the first row is 100 kHz.

SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:VIDeo

Sets or queries the resolution video bandwidth (VBW) value in the specified row in the Offsets & Limits table for the LTE ACLR display.

Conditions Measurement view: LTE ACLR

The parameter <x> represents the row in the Offsets & Limits table.

Group Sense commands

Syntax
```
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:VIDeo <value>
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:VIDeo?
```

Arguments <value>::<NRf> specifies the VBW in the specifies row.

Returns <NR3>

Examples `SENSE:LTE:ACLR:OFFSET1:BANDWIDTH:RESOLUTION:VIDEO 40.000000000E+3` sets the resolution VBW to 40 kHz for row 1.

SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:VIDeo:STATe

Sets or queries whether to enable or disable the video resolution bandwidth (VBW) in the specified row in the Offsets & Limits table for the LTE ACLR display.

Conditions Measurement view: LTE ACLR

The parameter <x> represents the row in the Offsets & Limits table.

Group Sense commands

Syntax

```
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:VIDeo:STATe
{ 0 | 1 }
SENSe:LTE:ACLR:OFFSet<x>:BANDwidth:RESolution:VIDeo:STATe?
```

Arguments 0 disables the VBW.

1 enables the VBW.

Returns 0 means the resolution VBW is disabled in the specified row.

1 means the VBW is enabled in the specified row.

Examples `SENSE:LTE:ACLR:OFFSET1:BANDWIDTH:RESOLUTION:VIDEO:STATE 1` enables the resolution VBW for row 1.

SENSe:LTE:ACLR:OFFSet<x>:DETection

Sets or queries the detection settings for all offsets for the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Sense commands

Syntax

```
SENSe:LTE:ACLR:OFFSet<x>:DETection { AVERage | PEAK }
SENSe:LTE:ACLR:OFFSet<x>:DETection?
```

Arguments AVERage: sets the offsets setting Average (VRMS)

PEAK: sets the offsets setting is the positive peak

Returns AVERage means the offsets setting Average (VRMS)

PEAK means the offsets setting is the positive peak

Examples `SENSE:LTE:ACLR:OFFSET<x>:DETECTION AVER` sets the detection settings for all offsets to Average RMS.

SENSe:LTE:ACLR:OFFSet<x>:FREQuency:BANDwidth

Sets or queries the frequency bandwidth value for the specified row of the Offsets & Limits table for the LTE ACLR display.

Conditions Measurement view: LTE ACLR

The parameter <x> represents the row in the Offsets & Limits table.

Group Sense commands

Syntax
```
SENSe:LTE:ACLR:OFFSet<x>:FREQuency:BANDwidth <value>
SENSe:LTE:ACLR:OFFSet<x>:FREQuency:BANDwidth?
```

Arguments <value>::<NRf> specifies the frequency bandwidth value

Returns <NR3>

Examples `SENSE:LTE:ACLR:OFFSET1:FREQUENCY:BANDWIDTH?` might return 4.5150000000E+6, indicating frequency bandwidth value of row 1 in the Offset & Limits table.

SENSe:LTE:ACLR:OFFSet<x>:FREQuency:OFFSet

Sets or queries the frequency offset value of the specified row in the Offsets & Limits table for the LTE ACLR display.

Conditions Measurement view: LTE ACLR

The parameter <x> represents the row in the Offsets & Limits table.

Group Sense commands

Syntax

```
SENSe:LTE:ACLR:OFFSet<x>:FREQuency:OFFSet <value>
SENSe:LTE:ACLR:OFFSet<x>:FREQuency:OFFSet?
```

Arguments <value>::<NRf> specifies the frequency offset value

Returns <NR3>

Examples `SENSE:LTE:ACLR:OFFSET1:FREQUENCY:OFFSET?` might return 5.0000000000E+6 indicating frequency offset value of row 1 in the Offset & Limits table tab.

SENSe:LTE:ACLR:OFFSet<x>:LIMit:ABSolute

Sets or queries the absolute limit value of the row in the Offset & Limit table tab for the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Parameter <x> represents the row in the Offsets & Limits table

Group Sense commands

Syntax

```
SENSe:LTE:ACLR:OFFSet<x>:LIMit:ABSolute <value>
SENSe:LTE:ACLR:OFFSet<x>:LIMit:ABSolute?
```

Arguments <value>::<NRf> specifies the absolute limit value of the specified row in the Offset & Limit table.

Returns <NRf>

Examples `SENSE:LTE:ACLR:OFFSET1:LIMIT:ABSOLUTE?` might return –6.4500000000 indicating the absolute limit value of row 1 is –6.4500000000 dBm.

SENSe:LTE:ACLR:OFFSet<x>:LIMit:MASK

Sets or queries the limit mask setting for the specified row in the Offsets & Limits table for the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Parameter <x> represents the row in the Offsets & Limits table

Group Sense commands

Syntax
```
SENSe:LTE:ACLR:OFFSet<x>:LIMit:MASK { ABS | REL | AND | OR }
SENSe:LTE:ACLR:OFFSet<x>:LIMit:MASK?
```

Arguments ABS specifies the limit mask is set to Absolute.

REL specifies the limit mask is set to Relative

AND specifies the limit mask is set to Absolute AND Relative.

OR specifies the limit mask is set to Absolute OR Relative.

Returns ABS means that the limit mask is set to Absolute.

REL means that the limit mask is set to Relative

AND means that the limit mask is set to Absolute AND Relative.

OR means that the limit mask is set to Absolute OR Relative.

Examples `SENSE:LTE:ACLR:OFFSET3:LIMIT:MASK REL` sets the limit mask for row 3 to Relative.

SENSe:LTE:ACLR:OFFSet<x>:LIMit:RELative

Sets or queries the relative limit value of channel under Offset & Limit table tab for the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Parameter <x> represents the row in the Offsets & Limits table

Group Sense commands

Syntax
```
SENSe:LTE:ACLR:OFFSet<x>:LIMit:RELative <value>
SENSe:LTE:ACLR:OFFSet<x>:LIMit:RELative?
```

Arguments <value>::<NRf> specifies the relative limit value of the specified row in the Offset & Limit table.

Returns <NRf>

Examples `SENSE:LTE:ACLR:OFFSET1:LIMIT:RELATIVE?` might return –44.2000000000, indicating the relative limit value of row 1 is –44.2000000000 dB.

SENSe:LTE:ACLR:OFFSet<x>:LIMit:SIDE

Sets or queries the side of the offset limit for the specified row in the Offsets & Limits table for the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Parameter <x> represents the row in the Offsets & Limits table

Group Sense commands

Syntax
```
SENSe:LTE:ACLR:OFFSet<x>:LIMit:SIDE { LEFT | RIGHt | BOTH }
SENSe:LTE:ACLR:OFFSet<x>:LIMit:SIDE?
```

Arguments LEFT specifies the left offset limit.

RIGHt specifies the right offset limit.

BOTH specifies both sides of the offset limit.

Returns LEFT means the left side offset limit is used for the specified row.

RIGHt means the right side offset limit is used for the specified row.

BOTH means both sides of the offset limit are used for the specified row.

Examples `SENSE:LTE:ACLR:OFFSET1:LIMIT:SIDE BOTH` specifies the measurement uses both sides of the offset in row 1.

SENSe:LTE:ACLR:OFFSet<x>:STATe

Sets or queries the specified row of the Offsets & Limits table for the LTE ACLR display.

Conditions Measurement view: LTE ACLR

The parameter <x> represents the row in the Offsets & Limits table.

Group Sense commands

Syntax
```
SENSe:LTE:ACLR:OFFSet<x>:STATe { OFF | ON | 0 | 1 }
SENSe:LTE:ACLR:OFFSet<x>:STATe?
```

Arguments OFF or 0 turns off display of the row in the measurement view.

ON or 1 turns on display of the row in the measurement view.

Returns OFF or 0 means the display of the row in the measurement view is off.

ON or 1 means the display of the row in the measurement view is on.

Examples `SENSE:LTE:ACLR:OFFSET2:STATE OFF` means that the display of row 2 in the Offsets & Limits table.

SENSe:LTE:ACLR:POINts:COUNt

Sets or queries the number of points used for the trace in the LTE ACLR measurement.

Conditions Measurement view: LTE ACLR

Group Sense commands

Syntax
```
SENSe:LTE:ACLR:POINts:COUNt <value>
SENSe:LTE:ACLR:POINts:COUNt?
```

Arguments <value>::<NRf> specifies the number of points used for the trace

Returns P<NR1>

Examples `SENSE:LTE:ACLR:POINTS:COUNT?` might return P8001, indicating a maximum of 8001 trace points can be used.

SENSe:LTE:ACLR:REFerence:MAGNitude

Sets or queries the reference power magnitude for the LTE ACLR measurement.

Conditions Measurement view: LTE ACLR

Group Sense commands

Syntax

```
SENSe:LTE:ACLR:REFerence:MAGNitude <value>
SENSe:LTE:ACLR:REFerence:MAGNitude?
```

Arguments <value>::<NRf> specifies the reference power magnitude

Returns <NRf>

Examples `SENSE:LTE:ACLR:REFERENCE:MAGNITUDE?` might return a –51.057, indicating that the reference power magnitude is –51.057 dBm.

SENSe:LTE:ACLR:REFerence:MAGNitude:AUTO

Sets or queries the reference magnitude to automatic or manual for the LTE ACLR measurement.

Conditions Measurement view: LTE ACLR

Group Sense commands

Syntax

```
SENSe:LTE:ACLR:REFerence:MAGNitude:AUTO { 0 | 1 }
SENSe:LTE:ACLR:REFerence:MAGNitude:AUTO?
```

Arguments 0 sets reference magnitude to Manual.

1 sets reference magnitude to Automatic.

Returns 0 means the reference magnitude is set to Manual.

1 means the reference magnitude is set to Automatic.

Examples `SENSE:LTE:ACLR:REFERENCE:MAGNITUDE:AUTO 0` sets the reference magnitude to Manual.

SENSe:LTE:ACLR:REFerence:MAGNitude:DETection

Sets or queries the reference magnitude detection for the LTE ACLR measurement.

Conditions Measurement view: LTE ACLR

Group Sense commands

Syntax
```
SENSe:LTE:ACLR:REFerence:MAGNitude:DETection { AVERage | PEAK }
SENSe:LTE:ACLR:REFerence:MAGNitude:DETection?
```

Arguments AVER sets reference magnitude detection to Avg(VRMS)

PEAK sets reference magnitude detection to +Peak.

Returns AVER means the reference magnitude detection is set to Avg(VRMS)

PEAK means the reference magnitude detection is set to +Peak.

Examples `SENSE:LTE:ACLR:REFERENCE:MAGNITUDE:DETECTION AVER` sets the magnitude detection to Avg(RMS).

SENSe:LTE:ACLR:STEP:AUTO

Determines whether to automatically update the center frequency increment/decrement step size in the LTE ACLR display. The step value controls the magnitude of each step when using the front panel controls or arrow keys on the keyboard.

Conditions Measurement view: LTE ACLR

Group Sense commands

Syntax
```
SENSe:LTE:ACLR:STEP:AUTO { OFF | ON | 0 | 1 }
SENSe:LTE:ACLR:STEP:AUTO?
```

Arguments OFF or 0 turns off the automatic update.

ON or 1 turns on automatic updates center frequency.

Returns OFF or 0 means automatic update is off.

ON or 1 means automatic updates are off.

Examples `SENSE:LTE:ACLR:STEP:AUTO ON` automatically updates the center frequency.

SENSe:LTE:ACLR:STEP:MAGNitude

Sets or queries the step magnitude for the center frequency in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Sense commands

Syntax
```
SENSe:LTE:ACLR:STEP:MAGNitude <value>
SENSe:LTE:ACLR:STEP:MAGNitude?
```

Arguments <value>::<NRf> specifies the center frequency

Returns <NR3>

Examples `SENSE:LTE:ACLR:STEP:MAGNITUDE 4.000000000E+6` sets the step magnitude to 4 MHz.

SENSe]:LTE:ANALysis:EQUAlization:STATe

Sets or queries the Enable Equalization state in the Analysis Params tab of the LTE Settings control panel.

Conditions Measurement view: LTE Constellation, LTE Power vs Time

Group Sense commands

Syntax

```
SENSe]:LTE:ANALysis:EQUAlization:STATe { 0 | 1 }
SENSe]:LTE:ANALysis:EQUAlization:STATe?
```

Arguments

0 disables the Enable Equalization.

1 enables the Enable Equalization.

Returns

0 means that Enable Equalization is disabled.

1 means that Enable Equalization is enabled.

Examples

`SENSE]:LTE:ANALYSIS:EQUALIZATION:STATE?` might return 0, indicating that Enable Equalization is disabled.

SENSe:LTE:ANALysis:LENGth:ACTual? (Query Only)

Queries the actual analysis length for the LTE Constellation and LTE Power vs Time displays. A query returns a value in either seconds or symbols.

Conditions

Measurement view: LTE Constellation, LTE Power vs Time

Group

Sense commands

Syntax

```
SENSe:LTE:ANALysis:LENGth:ACTual?
```

Returns

<NRf>

Examples

`SENSE:LTE:ANALYSIS:LENGTH:ACTUAL?` might return 43, indicating that the actual analysis length is 43 symbols.

[SENSe]:LTE:ANALysis:OFFSet

Sets or queries the LTE analysis offset for the LTE Constellation or LTE Power vs Time measurements. The query command returns the result in symbols or seconds, depending on the unit choice selected in the Analysis Time tab of the LTE Settings control panel.

Conditions

Measurement view: LTE Constellation, LTE Power vs Time

Group Sense commands

Syntax `[SENSe]:LTE:ANALysis:OFFSet <value>`

Arguments <value> specifies the analysis offset in the Analysis Time tab of the LTE Settings control panel.

Returns <NRf> Units are determined from the units choice selected in the Analysis Time tab of the LTE Settings control panel.

Examples `[SENSE]:LTE:ANALYSIS:OFFSET` –50 sets the analysis offset to –50 symbols.

SENSe:LTE:ANALysis:OFFSet:AUTO

Sets or queries the Auto checkbox state for analysis offset. This setting determines if the analysis offset for the LTE Constellation and LTE Power vs Time displays is set automatically or manually.

Conditions Measurement view: LTE Constellation, LTE Power vs Time

Group Sense commands

Syntax `SENSe:LTE:ANALysis:OFFSet:AUTO { 0 | 1 }`

Arguments 0 sets the analysis offset to manual (box is unchecked)

1 sets the analysis offset to automatic (checkbox is checked)

Returns 0 means the analysis offset is set to manual (box is unchecked)

1 means the analysis offset is set to automatic (checkbox is checked)

Examples `SENSE:LTE:ANALYSIS:OFFSET:AUTO 1` sets the analysis offset to automatic.

SENSe:LTE:ANALysis:TIMe:UNIT

Sets or queries the unit preference under the Analysis tab of the LTE Settings control panel for the LTE Constellation and LTE Power vs Time displays.

Conditions Measurement view: LTE Constellation, LTE Power vs Time

Group Sense commands

Syntax
```
SENSe:LTE:ANALysis:TIMe:UNIT { SEC | SYMB }
SENSe:LTE:ANALysis:TIMe:UNIT?
```

Arguments SEC sets the units to seconds.

SYMB sets the units to symbols.

Returns SEC means the units are set to seconds.

SYMB means the units are set to symbols.

Examples `SENSE:LTE:ANALYSIS:TIME:UNIT SEC` sets the units to seconds.

SENSe:LTE:CHANnel:BANDwidth

Specifies or queries the channel bandwidth to use for the LTE Constellation or LTE Power vs Time measurements.

Conditions Measurement view: LTE Constellation, LTE Power vs Time

Group Sense commands

Syntax `SENSe:LTE:CHANnel:BANDwidth { BW5M | BW10M | BW15M | BW20M }`

Arguments BW5M, BW10M, BW15M, BW20M each specifies a channel bandwidth to use for the measurement.

Returns Query returns one of the argument values, indicating the selected channel bandwidth.

Examples `SENSE:LTE:CHANNEL:BANDWIDTH BW5M` sets the channel bandwidth to use to 5 MHz.

SENSe:LTE:CHSPectrum:AVERage

Sets or queries how to average waveform in the LTE Channel Spectrum display.

Conditions Measurement view: LTE Channel Spectrum

Group Sense commands

Syntax

```
SENSe:LTE:CHSPectrum:AVERage { OFF | TIME | FREQ }
SENSe:LTE:CHSPectrum:AVERage?
```

Arguments OFF disables averaging.

TIME averages the waveform using time samples.

FREQuency averages the waveform using frequency samples.

Returns OFF means averaging is disabled.

TIME means averaging of the waveform is performed using time samples.

FREQuency means averaging of the waveform is performed using frequency samples.

Examples `SENSE:LTE:CHSPECTRUM:AVERAGE TIME` sets the analyzer to average the waveform using time samples.

SENSe:LTE:CHSPectrum:AVERage:COUNt

Sets or queries the number of traces for averaging in the LTE Channel Spectrum display.

Conditions Measurement view: LTE Channel Spectrum

Group Sense commands

Syntax

```
SENSe:LTE:CHSPectrum:AVERage:COUNt <number>
SENSe:LTE:CHSPectrum:AVERage:COUNt?
```

Arguments <number>::<NR1> specifies the average count

Returns <NR1>

Examples SENSE:LTE:CHSPECTRUM:AVERAGE:COUNT? might return 10, indicating that 10 is the number of traces for averaging.

SENSe:LTE:CHSPectrum:BANDwidth:RESolution

Sets or queries the resolution bandwidth (RBW) in the LTE Channel Spectrum display.

Conditions Measurement view: LTE Channel Spectrum

Group Sense commands

Syntax SENSe:LTE:CHSPectrum:BANDwidth:RESolution

Arguments <value>::<NRf> specifies the resolution bandwidth

Returns <NR3>

Examples SENSE:LTE:CHSPECTRUM:BANDWIDTH:RESOLUTION? might return 30.0000000E+3, indicating the RBW is 30 kHz.

SENSe:LTE:CHSPectrum:BANDwidth:RESOlution:AUTO

Determines whether to set the resolution bandwidth (RBW) in the LTE Channel Spectrum display automatically or manually.

Conditions Measurement view: LTE Channel Spectrum

Group Sense commands

Syntax
```
SENSe:LTE:CHSPectrum:BANDwidth:RESOlution:AUTO { OFF | ON
| 0 | 1 }
SENSe:LTE:CHSPectrum:BANDwidth:RESOlution:AUTO?
```

Arguments OFF or 0 sets the RBW to manual.

ON or 1 sets the RBW to automatic.

Returns OFF or 0 means that the RBW is set to manual.

ON or 1 means that the RBW is set to automatic.

Examples `SENSE:LTE:CHSPECTRUM:BANDWIDTH:RESOLUTION:AUTO 1` sets the RBW to automatic and is decided by the analysis module.

SENSe:LTE:CHSPectrum:BANDwidth:VIDeo

Sets or queries the video bandwidth (VBW) in the LTE Channel Spectrum display.

Conditions Measurement view: LTE Channel Spectrum

Group Sense commands

Syntax `SENSe:LTE:CHSPectrum:BANDwidth:VIDeo <value>`

Arguments <value>::<NRf> specifies the video bandwidth

Returns <NR3>

Examples `SENSE:LTE:CHSPECTRUM:BANDWIDTH:VIDEO 3.0000000E+3` sets the VBW to 3 kHz.

SENSe:LTE:CHSPectrum:BANDwidth:VIDeo:STATe

Sets or queries enabling or disabling the video bandwidth (VBW) in the LTE Channel Spectrum display.

Conditions Measurement view: LTE Channel Spectrum

Group Sense commands

Syntax

```
SENSe:LTE:CHSPectrum:BANDwidth:VIDeo:STATe { 0 | 1 }
SENSe:LTE:CHSPectrum:BANDwidth:VIDeo:STATe?
```

Arguments

0 disables the VBW.

1 enables the VBW.

Returns

0 means that the VBW is disabled.

1 means that the VBW is enabled.

Examples

`SENSE:LTE:CHSPECTRUM:BANDWIDTH:VIDEO:STATE 1` sets the VBW to enabled.

SENSe:LTE:CHSPectrum:CHANnel:BANDwidth

Specifies or queries a nominal channel bandwidth to use for the LTE Channel Spectrum measurements.

Conditions

Measurement view: LTE Channel Spectrum

Group

Sense commands

Syntax

```
SENSe:LTE:CHSPectrum:CHANnel:BANDwidth { BW5M | BW10M |
BW20M | BW1PT4M | BW3M | BW15M }
```

Arguments

BW5M, BW10M, BW20M, BW1PT4M, BW3M, BW15M each specifies a nominal channel bandwidth to use for the measurement.

Returns

Query returns one of the argument values, indicating the selected nominal channel bandwidth.

Examples

`SENSE:LTE:CHSPECTRUM:CHANNEL:BANDWIDTH BW20M` sets the nominal channel bandwidth to use to 20 MHz.

SENSe:LTE:CHSPectrum:CLEar:RESults (No Query Form)

Clears the results in the LTE Channel Spectrum measurement.

Conditions Measurement view: LTE Channel Spectrum

Group Sense commands

Syntax `SENSe:LTE:CHSPectrum:CLEar:RESults`

Arguments None

Examples `SENSE:LTE:CHSPECTRUM:CLEAR:RESULTS` clears the results of the LTE Channel Spectrum measurement.

SENSe:LTE:CHSPectrum:DETection

Sets or queries the trace detection settings for the LTE Channel Spectrum display.

Conditions Measurement view: LTE Channel Spectrum

Group Sense commands

Syntax
```
SENSe:LTE:CHSPectrum:DETection { POS | AVER }
SENSe:LTE:CHSPectrum:DETection?
```

Related Commands

Arguments AVERage: sets the detection setting to Average (VRMS)

POS: sets the detection setting to the positive peak

Returns AVERage means the detection setting is Average (VRMS)

POS means the detection setting is positive peak

Examples `SENSE:LTE:CHSPECTRUM:DETECTION POS` sets the trace detection settings to +Peak under the Prefs tab.

SENSe:LTE:CHSPectrum:FREQuency

Sets or queries the measurement center frequency in the LTE Channel Spectrum display.

Conditions Measurement view: LTE Channel Spectrum

Group Sense commands

Syntax
```
SENSe:LTE:CHSPectrum:FREQuency <value>
SENSe:LTE:CHSPectrum:FREQuency?
```

Arguments <value>::<NRf> specifies the center frequency

Returns <NR3>

Examples `SENSE:LTE:CHSPECTRUM:FREQUENCY` 1.000000000E+9 sets the center frequency to 1.0 GHz.

SENSe:LTE:CHSPectrum:FREQuency:SPAN

Sets or queries the frequency span in the LTE Channel Spectrum measurement.

Conditions Measurement view: LTE Channel Spectrum

Group Sense commands

Syntax
```
SENSe:LTE:CHSPectrum:FREQuency:SPAN <value>
SENSe:LTE:CHSPectrum:FREQuency:SPAN?
```

Arguments <value>::<NRf> specifies the frequency span

Returns <NRf>

Examples `SENSE:LTE:CHSPECTRUM:FREQUENCY:SPAN` sets the frequency span in the LTE Channel Spectrum display.

SENSe:LTE:CHSPectrum:FREQuency:STEP

Sets or queries the frequency step size in the LTE Channel Spectrum measurement. Programming a specified step size sets `SENSe:LTE:CHSPectrum:FREQuency:STEP:AUTO` to `OFF`.

Conditions Measurement view: LTE Channel Spectrum

Group Sense commands

Syntax
```
SENSe:LTE:CHSPectrum:FREQuency:STEP <value>
SENSe:LTE:CHSPectrum:FREQuency:STEP?
```

Arguments <value>::<NRf> specifies the frequency step size

Returns <NR3>

Examples `SENSE:LTE:CHSPECTRUM:FREQUENCY:STEP 200.000000000E+3` sets the frequency step size to 200 kHz.

SENSe:LTE:CHSPectrum:FREQuency:STEP:AUTO

Determines whether to set the frequency step size automatically or manually in the LTE Channel Spectrum measurement.

Conditions Measurement view: LTE Channel Spectrum

Group Sense commands

Syntax
```
SENSe:LTE:CHSPectrum:FREQuency:STEP:AUTO { OFF | ON | 0 |
1 }
SENSe:LTE:CHSPectrum:FREQuency:STEP:AUTO?
```

Arguments OFF or 0 sets the frequency step size is set to manual.

ON or 1 sets the frequency step size is set to automatic.

Returns OFF or 0 means that the frequency step size is set to manual.

ON or 1 means that the frequency step size is set to automatic.

Examples `SENSE:LTE:CHSPECTRUM:FREQUENCY:STEP:AUTO ON` specifies that the frequency step size is set automatically.

SENSe:LTE:CHSPectrum:POINts:COUNt

Sets or queries the number of points used for the trace in the LTE Channel Spectrum measurement.

Conditions Measurement view: LTE Channel Spectrum

Group Sense commands

Syntax
```
SENSe:LTE:CHSPectrum:POINts:COUNt <value>
SENSe:LTE:CHSPectrum:POINts:COUNt?
```

Arguments <value>::<NRf> specifies the number of points used for the trace

Returns P<NR1>

Examples `SENSE:LTE:CHSPECTRUM:POINTS:COUNT?` might return P8001, indicating a maximum of 8001 trace points can be used.

SENSe:LTE:FRAMe:STRUcture

Sets or queries the frame structure in the Modulation Params tab of the LTE settings control panel.

Conditions Measurement view: LTE Constellation and LTE Power vs Time

Group Sense commands

Syntax
```
SENSe:LTE:FRAMe:STRUcture { FDD | TDD }
SENSe:LTE:FRAMe:STRUcture?
```

Arguments FDD

TDD

Returns FDD

TDD

Examples `SENSE:LTE:FRAME:STRUCTURE FDD` sets the frame structure to FDD.

[SENSe]:MCPower:AVERage

Sets or queries the average method in the MCPR measurement.

Conditions Measurement views: MCPR

Group Sense commands

Syntax
```
[SENSe]:MCPower:AVERage { OFF | TIMe | FREQuency }
[SENSe]:MCPower:AVERage?
```

Arguments `OFF` disables averaging.

`TIMe` performs averaging for time samples.

`FREQuency` performs averaging for frequency samples.

Examples `SENSE:MCPOWER:AVERAGE TIMe` performs averaging for time samples.

[SENSe]:MCPower:AVERage:COUNt

Sets or queries the average count in the MCPR measurement.

Conditions Measurement views: MCPR

Group Sense commands

Syntax
```
[SENSe]:MCPower:AVERage:COUNt <value>
[SENSe]:MCPower:AVERage:COUNt?
```

Arguments `<value> ::  <NR1>` specifies the average count. Range: 2 to 10000.

Examples `SENSE:MCPOWER:AVERAGE:COUNT 256` sets the average count to 256.

[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution]

Sets or queries the resolution bandwidth (RBW). Programming a specified RBW sets [SENSe]:MCPower{BANDwidth|BWIDth}[:RESolution]:AUTO OFF.

Conditions Measurement views: MCPR

Group Sense commands

Syntax
```
[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution] <value>
[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution]?
```

Related Commands [SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution]:AUTO

Arguments `<value> ::` `<NRf>` specifies the RBW. Range: 100 Hz to 5 MHz.

Examples `SENSE:MCPOWER:BANDWIDTH:RESOLUTION 200kHz` sets the RBW to 200 kHz.

[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution]:ACTual? (Query Only)

Queries the actual resolution bandwidth (RBW) in the MCPR measurement.

Conditions Measurement views: MCPR

Group Sense commands

Syntax
```
[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution]:ACTual?
```

Arguments None

Returns `<NRf>` The actual RBW in Hz.

Examples `SENSE:MCPOWER:BANDWIDTH:RESOLUTION:ACTUAL?` might return `299.624E+3`, indicating that the actual RBW is 299.624 kHz.

[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution]:AUTO

Determines whether to set the resolution bandwidth (RBW) automatically or manually.

Conditions Measurement views: MCPR

Group Sense commands

Syntax

```
[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution]:AUTO { OFF | ON | 0 | 1 }
[SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution]:AUTO?
```

Arguments `OFF` or `0` specifies that the resolution bandwidth is set manually using the [SENSe]:MCPower:{BANDwidth|BWIDth}[:RESolution] command.

`ON` or `1` specifies that the resolution bandwidth is set automatically.

Examples `SENSE:MCPOWER:BANDWIDTH:AUTO ON` sets the resolution bandwidth automatically.

[SENSe]:MCPower:{BANDwidth|BWIDth}:VIDeo

Sets or queries the video bandwidth (VBW). Programming a specified VBW sets [SENSe]:MCPower:{BANDwidth|BWIDth}:VIDeo:STATe OFF.

Conditions Measurement views: MCPR

Group Sense commands

Syntax

```
[SENSe]:MCPower:{BANDwidth|BWIDth}:VIDeo <value>
[SENSe]:MCPower:{BANDwidth|BWIDth}:VIDeo?
```

Related Commands [SENSe]:MCPower:{BANDwidth|BWIDth}:VIDeo:STATe

Arguments `<value> :: <NRf>` specifies the VBW.
Range: Current RBW/10^4 (1 Hz minimum) to Current RBW.

Examples `SENSE:MCPOWER:BANDWIDTH:VIDEO 200kHz` sets the VBW to 200 kHz.

[SENSe]:MCPower:{BANDwidth|BWIDth}:VIDeo:STATe

Determines whether to enable the video bandwidth (VBW) in the MCPR measurement.

Conditions Measurement views: MCPR

Group Sense commands

Syntax

```
[SENSe]:MCPower:{BANDwidth|BWIDth}:VIDeo:STATe { OFF | ON | 0 | 1 }
[SENSe]:MCPower:{BANDwidth|BWIDth}:VIDeo:STATe?
```

Arguments `OFF` or `0` disables the VBW.

`ON` or `1` enables the VBW.

Examples `SENSE:MCPOWER:BANDWIDTH:VIDEO:STATE ON` enables the VBW.

[SENSe]:MCPower:CHANnel:ADJacent:ADD (No Query Form)

Adds a pair of upper and lower adjacent channels in the MCPR measurement.

Conditions Measurement views: MCPR

Group Sense commands

Syntax

```
[SENSe]:MCPower:CHANnel:ADJacent:ADD <offset>,<bandwidth>
```

Arguments `<offset> ::` `<NRf>` specifies the offset from the center frequency for the adjacent channel.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

`<bandwidth> ::` `<NRf>` specifies the bandwidth of the adjacent channel.

Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `SENSE:MCPOWER:CHANNEL:ADJACENT:ADD 200kHz,80kHz` adds a pair of upper and lower adjacent channels with the offset of ±200 kHz and the bandwidth of 80 kHz.

[SENSe]:MCPower:CHANnel:ADJacent:DELete (No Query Form)

Deletes a selected adjacent channel in the MCPR measurement.

Conditions Measurement views: MCPR

Group Sense commands

Syntax `[SENSe]:MCPower:CHANnel:ADJacent:DELete <channel>`

Arguments `<channel> :: <string>` specifies the channel to be deleted.
Specify the channel with `"A<n>"` for the adjacent channel where <n> represents the channel number (<n> = 1, 2, 3,...). See the example below.

Examples `SENSe:MCPower:CHANnel:ADJacent:DELete "A2"` deletes A2 (the adjacent channel 2).

[SENSe]:MCPower:CHANnel:ADJacent:LIMIts<x>

This command returns the selected adjacent upper and lower limit values. Selection of adjacent channel.

Conditions Measurement view: P25 MCPR

The parameter <x>=1 to 100.

Group Sense commands

Syntax

```
[SENSe]:MCPower:CHANnel:ADJacent:LIMIts<x>
[SENSe]:MCPower:CHANnel:ADJacent:LIMIts<x>?
```

Arguments <x>=1 to 100

Returns Upper and lower limit values for selected adjacent channels.

Examples `SENSe:MCPower:CHANnel:ADJacent:LIMIts2?` returns upper and lower limit values of A2 (Adjacent channel 2).

[SENSe]:MCPower:CHANnel:ADJacent:LIMIts<x>:ENABle? (Query Only)

This query returns whether the adjacent channel is selected or not.

Conditions Measurement view: P25 MCPR

Group Sense commands

Syntax

```
[SENSe]:MCPower:CHANnel:ADJacent:LIMIts<x>:ENABle?
```

Returns `0` means the adjacent channel is disabled.

`1` means the adjacent channel is enabled.

Examples `SENSe:MCPower:CHANnel:ADJacent:LIMIts1:ENABle?` `1` means that A1 (adjacent channel 1) is enabled.

[SENSe]:MCPower:CHANnel:FILTer

Sets or queries the adjacent channel filter in the MCPR measurement.

Conditions Measurement views: MCPR

Group Sense commands

Syntax

```
[SENSe]:MCPower:CHANnel:FILTer { RRCosine | NONE }
[SENSe]:MCPower:CHANnel:FILTer?
```

Arguments RRCosine selects the Root-Raised-Cosine filter.

NONE uses no filter.

Examples SENSE:MCPOWER:CHANNEL:FILTER RRCosine selects Root-Raised-Cosine for the adjacent channel filter.

[SENSe]:MCPower:CHANnel:MAIN:{BANDwidth|BWIDth}

Sets or queries the frequency bandwidth of the main channels (all share the same value) in the MCPR measurement.

Conditions Measurement views: MCPR

Group Sense commands

Syntax
```
[SENSe]:MCPower:CHANnel:MAIN:{BANDwidth|BWIDth} <value>
[SENSe]:MCPower:CHANnel:MAIN:{BANDwidth|BWIDth}?
```

Arguments <value> :: <NRf> specifies the main channel bandwidth.
Range: 1 Hz to full span.

Examples SENSE:MCPOWER:CHANNEL:MAIN:BANDWIDTH 4.5MHz sets the main channel bandwidth to 4.5 MHz.

[SENSe]:MCPower:CHANnel:MAIN:COUNt

Sets or queries the number of main channels in the MCPR measurement. You can use this command to add and remove main channels.

Conditions Measurement views: MCPR

Group Sense commands

Syntax
```
[SENSe]:MCPower:CHANnel:MAIN:COUNt <value>
[SENSe]:MCPower:CHANnel:MAIN:COUNt?
```

Arguments <value> :: <NRf> specifies the number of main channels. Range: 1 to 99.

Examples `SENSE:MCPOWER:CHANNEL:MAIN:COUNT 3` sets the the number of main channels to 3.

[SENSe]:MCPower:CHANnel:MAIN:INACtive

Makes a specified main channel inactive. You can set it on or off. The query returns all inactive main channels.

Conditions Measurement views: MCPR

Group Sense commands

Syntax

```
[SENSe]:MCPower:CHANnel:MAIN:INACtive <channel>,<boolean>
[SENSe]:MCPower:CHANnel:MAIN:INACtive?
```

Arguments `<channel> :: <string>` specifies the channel to be inactive.
Specify the channel with "M<n>" for the main channel where <n> represents the channel number (<n> = 1, 2, 3,...). See the example below.

`<boolean>::={ OFF | ON | 0 | 1 }` specifies that the specified channel is inactive (On) or not (Off).

Examples `SENSE:MCPOWER:CHANNEL:MAIN:INACTIVE "M2",ON` makes the main channel 2 inactive.

[SENSe]:MCPower:CHANnel:MAIN:SPACing

Sets or queries frequency difference between centers of each main channel in the MCPR measurement.

Conditions Measurement views: MCPR

Group Sense commands

Syntax

```
[SENSe]:MCPower:CHANnel:MAIN:SPACing <value>
[SENSe]:MCPower:CHANnel:MAIN:SPACing?
```

Arguments `<value> :: <NRf>` specifies the spacing between two adjacent main channels. Range: 1 Hz to 1 GHz.

Examples `SENSE:MCPOWER:CHANNEL:MAIN:SPACING 5MHz` sets the main channel spacing to 5 MHz.

[SENSe]:MCPower:CHIPrate

Sets or queries the chip rate in the MCPR measurement. This command is valid when [SENSe]:MCPower:CHANnel:FILTer is set to RRCosine (Root-Raised-Cosine).

Conditions Measurement views: MCPR

Group Sense commands

Syntax
```
[SENSe]:MCPower:CHIPrate <value>
[SENSe]:MCPower:CHIPrate?
```

Arguments `<value> ::` `<NRf>` specifies the chip rate. Range: 100 Hz to 105 MHz.

Examples `SENSE:MCPOWER:CHIPRATE 1kHz` sets the chip rate to 1 kHz.

[SENSe]:MCPower:CLEar:RESults (No Query Form)

Restarts the average trace.

Conditions Measurement views: MCPR

Group Sense commands

Syntax `[SENSe]:MCPower:CLEar:RESults`

Arguments None

Examples `SENSE:MCPOWER:CLEAR:RESULTS` restarts the average trace.

[SENSe]:MCPower:FREQuency

Sets or queries the center frequency in the MCPR measurement.

Conditions Measurement views: MCPR

Group Sense commands

Syntax
```
[SENSe]:MCPower:FREQuency <value>
[SENSe]:MCPower:FREQuency?
```

Arguments <value> :: <NRf> specifies the center frequency.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B– 1 to 26.5 GHz

Examples SENSE:MCPOWER:FREQUENCY 2.35GHz sets the center frequency to 2.35 GHz.

[SENSe]:MCPower:FREQuency:STEP

Sets or queries the frequency step size. Programming a specified step size sets [SENSe]:MCPower:FREQuency:STEP:AUTO OFF.

Conditions Measurement views: MCPR

Group Sense commands

Syntax
```
[SENSe]:MCPower:FREQuency:STEP <value>
[SENSe]:MCPower:FREQuency:STEP?
```

Related Commands [SENSe]:MCPower:FREQuency:STEP:AUTO

Arguments <value> :: <NRf> specifies the frequency step size.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106AB – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `SENSE:MCPOWER:FREQUENCY:STEP 50kHz` sets the frequency step size to 50 kHz.

[SENSe]:MCPower:FREQuency:STEP:AUTO

Determines whether to set the frequency step size automatically or manually in the MCPR measurement.

Conditions Measurement views: MCPR

Group Sense commands

Syntax
```
[SENSe]:MCPower:FREQuency:STEP:AUTO { OFF | ON | 0 | 1 }
[SENSe]:MCPower:FREQuency:STEP:AUTO?
```

Arguments `OFF` or `0` specifies that the frequency step size is set manually using the [SENSe]:MCPower:FREQuency:STEP command.

`ON` or `1` specifies that the frequency step size is set automatically.

Examples `SENSE:MCPOWER:FREQUENCY:STEP:AUTO ON` specifies that the frequency step size is set automatically.

[SENSe]:MCPower:NFLoor:STATe

Determines whether to enable correction for noise floor.

Conditions Measurement views: MCPR

Group Sense commands

Syntax
```
[SENSe]:MCPower:NFLoor:STATe { OFF | ON | 0 | 1 }
[SENSe]:MCPower:NFLoor:STATe?
```

Arguments `OFF` or `0` disables correction for noise floor.

`ON` or `1` enables correction for noise floor.

Examples `SENSE:MCPOWER:NFLOOR:STATE ON` enables correction for noise floor.

[SENSe]:MCPower:POINts:COUNt

Sets or queries the number of trace points in the MCPR display.

Conditions Measurement views: MCPR

Group Sense commands

Syntax
```
[SENSe]:MCPower:POINts:COUNt { P801 | P2401 | P4001 | P8001
| P10801 }
[SENSe]:MCPower:POINts:COUNt?
```

Arguments `P801 | P2401 | P4001 | P8001 | P10801` set the number of trace points.

Examples `SENSE:MCPOWER:POINTS:COUNT P4001` sets the number of trace points in the MCPR measurement to 4001.

[SENSe]:MCPower:RCHannels? (Query Only)

Queries the power reference in the MCPR measurement.

Conditions Measurement views: MCPR

Group Sense commands

Syntax `[SENSe]:MCPower:RCHannels?`

Arguments None

Returns `<power_ref>::={ Total | M<x> }` where <x> = 1 to 99.

`Total` indicates that the power reference is the total power of all the active channels.

`M<x>` indicates that the power reference is the main channel with the index (<x>).

Examples `SENSE:MCPOWER:RCHANNELS?` might return `M3`, indicating that the power reference is the main channel 3.

[SENSe]:MCPower:RCHannels:MAIN<x> (No Query Form)

Sets the power reference to the main channel with the index (<x>) in the MCPR measurement.

The parameter <x> = 1 to 99, representing the main channel 1 to 99, respectively. The main channel must be defined using the [SENSe]:MCPower:CHANnel:MAIN commands.

Conditions Measurement views: MCPR

Group Sense commands

Syntax `[SENSe]:MCPower:RCHannels:MAIN<x>`

Related Commands [:SENSe]:MCPower:CHANnel:MAIN commands

Arguments None

Examples `SENSE:MCPOWER:RCHANNELS:MAIN3` selects Main 3 for the power reference channel.

[SENSe]:MCPower:RCHannels:TOTal (No Query Form)

Sets the power reference to the total power of all the active channels in the MCPR measurement.

Conditions Measurement views: MCPR

Group Sense commands

Syntax `[SENSe]:MCPower:RCHannels:TOTal`

Arguments None

Examples `SENSE:MCPOWER:RCHANNELS:TOTAL` sets the power reference to the total power of all the active channels.

[SENSe]:MCPower:RRCRolloff

Sets or queries the filter parameter (roll-off ratio) for the Root Raised Cosine filter.

Conditions Measurement views: MCPR

Group Sense commands

Syntax
```
[SENSe]:MCPower:RRCRolloff <value>
[SENSe]:MCPower:RRCRolloff?
```

Related Commands [SENSe]:MCPower:CHANnel:FILTer

Arguments `<value> ::  <NRf>` specifies the filter parameter.
Range: 0.001 to 1, 0.0001 step.

Examples `SENSE:MCPOWER:RRCROLLOFF 0.3` sets the filter parameter to 0.3.

[SENSe]:MEASurement:FREQuency

Sets or queries the measurement frequency.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:MEASurement:FREQuency <value>
[SENSe]:MEASurement:FREQuency?
```

Arguments `<value> ::  <NRf>` specifies the measurement frequency.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `SENSE:MEASUREMENT:FREQUENCY 7.5GHz` sets the measurement frequency to 7.5 GHz.

[SENSe]:MEASurement:FREQuency:CENTer:LOCK

Sets or queries locking the measurement center frequency.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:MEASurement:FREQuency:CENTer:LOCK { OFF | ON | 0 | 1 }
[SENSe]:MEASurement:FREQuency:CENTer:LOCK?
```

Arguments `OFF` or `0` disables locking the center frequency.

`ON` or `1` enables locking the center frequency.

Examples `[SENSE]:MEASUREMENT:FREQUENCY:CENTER:LOCK ON` locks the center frequency.

[SENSe]:NOISe:AVERage:COUNt

This command returns the acquisition counts over which the noise average is computed.

Conditions Measurement view: Noise Figure

Group Sense commands

Syntax

```
[SENSe]:NOISe:AVERage:COUNt <integer>
[SENSe]:NOISe:AVERage:COUNt?
```

Arguments

<integer>

Range: 1 to 1000

Returns

<NR3> = a floating point value with an exponent

Examples

SENSE:NOISE:AVERAGE:COUNT? might return 1.0000000000E+3, indicating the average count of the analyzer is set to 1000.

[SENSe]:NOISe:BANDwidth

This command sets or queries the resolution bandwidth (RBW) for the Noise Figure display measurement.

Conditions

Measurement view: Noise Figure

Group

Sense commands

Syntax

```
[SENSe]:NOISe:BANDwidth <NR3>
[SENSe]:NOISe:BANDwidth?
```

Arguments

<NR3> = a floating point value with an exponent

Range: 50 Hz - 10 MHz

Returns

<NR3> = a floating point value with an exponent

Examples

SENSE:NOISE:BANDWIDTH? might return 1.0000000000E+6, indicating that the RBW is 1.00 MHz.

[SENSE]:NOISe:CONTrol:SETTling:OFF[:TIME]

This command sets or queries the analyzer off settle time for the Noise Figure display measurement.

Conditions

Measurement view: Noise Figure

Group Sense commands

Syntax
```
[SENSE]:NOISe:CONTrol:SETTling:OFF[:TIME] <NR3>
[SENSE]:NOISe:CONTrol:SETTling:OFF[:TIME]?
```

Related Commands [SENSE]:NOISe:CONTrol:SETTling:ON[:TIME]

Arguments <NR3> = a floating point value with an exponent

Range: 0 s - 10 ks

Returns <NR3> = a floating point value with an exponent

Examples SENSE:NOISE:CONTROL:SETTLING:OFF:TIME? might return 200.0000000000E-3, indicating the the RSA settle time (off) is 200 ms.

[SENSE]:NOISe:CONTrol:SETTling:ON[:TIME]

This command sets or queries the analyzer on settle time for the Noise and Gain display measurements.

Conditions Measurement views: Noise Figure, Noise Temperature, Y Factor

Group Sense commands

Syntax
```
[SENSE]:NOISe:CONTrol:SETTling:ON[:TIME] <NR3>
[SENSE]:NOISe:CONTrol:SETTling:ON[:TIME]?
```

Related Commands [SENSE]:NOISe:CONTrol:SETTling:OFF[:TIME]

Arguments <NR3> = a floating point value with an exponent

Range: 0 s - 10 ks

Returns <NR3> = a floating point value with an exponent

Examples SENSE:NOISE:CONTROL:SETTLING:ON:TIME? might return 50.0000000000E-3, indicating the RSA settle time (on) is 50 ms.

[SENSe]:NOISe:CORRection:ENR:CONSTant

This command sets and queries the excess noise ratio (ENR) constant for the Noise Figure and Gain displays.

Conditions Measurement view: Noise Figure, Y Factor, Gain, Noise Temperature, Noise Table

Group Sense commands

Syntax
```
[SENSe]:NOISe:CORRection:ENR:CONSTant <NR2>
[SENSe]:NOISe:CORRection:ENR:CONSTant?
```

Related Commands [SENSe]:NOISe:CORRection:ENR:MODE

Arguments <NR2> = a floating point value without an exponent

Returns <NR2> = a floating point value without an exponent

Examples SENSE:NOISE:CORRECTION:ENR:CONSTANT? might return 10.0000000000, indicating that the ENR constant is 10.00 dB.

[SENSe]:NOISe:CORRection:ENR:MODE

This command sets and queries the Noise Source mode for the Noise Figure and Gain displays.

Conditions Measurement view: Noise Figure, Y Factor, Gain, Noise Temperature, Noise Table

Group Sense commands

Syntax
```
[SENSe]:NOISe:CORRection:ENR:MODE { TABLe | CONSTant }
[SENSe]:NOISe:CORRection:ENR:MODE?
```

Related Commands [SENSe]:NOISe:CORRection:ENR:CONSTant

Arguments

`TABLe` sets the noise source mode to Table.

`CONSTant` sets the noise source mode to Constant.

Returns

`TABL` indicates that the noise source mode is set to Table.

`CONST` indicates that the noise source mode is set to Constant.

Examples

`SENSE:NOISE:CORRECTION:ENR:MODE?` might return TABL, indicating that the noise source mode is set to Table.

[SENSe]:NOISe:CORRection:ENR:TABLe:DATA? (Query Only)

This command queries the Noise Source model, serial number, and table values for frequency and ENR for the Noise Figure and Gain displays.

Conditions

Measurement view: Noise Figure, Y Factor, Gain, Noise Temperature, Noise Table

Group

Sense commands

Syntax

`[SENSe]:NOISe:CORRection:ENR:TABLe:DATA?`

Related Commands

[SENSe]:NOISe:CORRection:ENR:MODE

[SENSe]:NOISe:CORRection:ENR:TABLe:ID

Returns

<string> is the Noise Source model and serial number, as well as the frequency and ENR values for the noise source table.

Examples

`SENSE:NOISE:CORRECTION:ENR:TABLE:DATA?` might return 346B 10MHz – 18 GHz, 2037A02333, 2900000000, 15, 2900000000, 15, 2900000000, 15, 2900000000, 15, 2900000000, 15, 2900000000, 15, 2900000000, 15, 2900000000, 15, 2900000000, 15, 2900000000, 15, 2900000000, 15. The first two values indicate the Noise Source model and serial numbers. All of the values that follow indicate the frequency (2900000000, meaning 2.90 GHz) and the ENR (15, meaning 15.00 dB) shown in the Noise Source table.

[SENSe]:NOISe:CORRection:ENR:TABLe:ID

This command sets or queries the Noise Source Model for the Noise Figure and Gain displays.

Conditions Measurement view: Noise Figure, Y Factor, Gain, Noise Temperature, Noise Table

Group Sense commands

Syntax
```
[SENSe]:NOISe:CORRection:ENR:TABLe:ID <Qstring>
[SENSe]:NOISe:CORRection:ENR:TABLe:ID?
```

Related Commands [SENSe]:NOISe:CORRection:ENR:TABLe:DATA?

Arguments <Qstring> = quoted string of ASCII text

Returns <Qstring> = quoted string of ASCII text

Examples `SENSE:NOISE:CORRECTION:ENR:TABLE:ID?` might return "346B 10MHz – 18 GHz", indicating the Noise Source model is 346B 10MHz–18 GHz.

[SENSe]:NOISe:CORRection:ENR:TABLe:NEW (No Query Form)

This command sets the Noise Source Model for the Noise Figure and Gain displays.

Conditions Measurement view: Noise Figure, Y Factor, Gain, Noise Temperature, Noise Table

Group Sense commands

Syntax
```
[SENSe]:NOISe:CORRection:ENR:TABLe:NEW <Qstring>
[SENSe]:NOISe:CORRection:ENR:TABLe:NEW?
```

Related Commands [SENSe]:NOISe:CORRection:ENR:TABLe:DATA?

Arguments <Qstring> = quoted string of ASCII text

Returns <Qstring> = quoted string of ASCII text

Examples `SENSE:NOISE:CORRECTION:ENR:TABLE:NEW` might return "346B 10MHz – 18 GHz", indicating the Noise Source model is 346B 10MHz–18 GHz.

[SENSe]:NOISe:CORRection:ENR:TABLe:SERial

This command sets or queries the Noise Source serial number for the Noise Figure and Gain displays.

Conditions Measurement view: Noise Figure, Y Factor, Gain, Noise Temperature, Noise Table

Group Sense commands

Syntax
```
[SENSe]:NOISe:CORRection:ENR:TABLe:SERial <Qstring>
[SENSe]:NOISe:CORRection:ENR:TABLe:SERial?
```

Related Commands [SENSe]:NOISe:CORRection:ENR:TABLe:DATA?

Arguments <Qstring> = quoted string of ASCII text

Returns <Qstring> = quoted string of ASCII text

Examples `SENSE:NOISE:CORRECTION:ENR:TABLE:SERIAL?` might return "2037A02334", indicating the Noise Source serial number is 2037A02334.

[SENSe]:NOISe:CORRection:TCOLd:ADUT:UREFerence

Sets or queries the analyzer to use the reference temperature (K) for the loss at the DUT output in the noise displays.

Conditions Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group Sense commands

Syntax

```
[SENSe]:NOISe:CORRection:TCOLd:ADUT:UREFerence { OFF | ON
| 1 | 0 }
[SENSe]:NOISe:CORRection:TCOLd:ADUT:UREFerence?
```

Arguments

OFF or 0 disables the Use Ref temperature feature for the Loss at DUT Output.

ON or 1 enables the Use Ref temperature feature for the Loss at DUT Output.

Returns

0 means that the Use Ref temperature feature for the Loss at DUT Output is disabled.

1 means that the Use Ref temperature feature for the Loss at DUT Output is enabled.

Examples

`SENS:NOIS:CORR:TCOL:ADUT:UREF?` might return 1, indicating that the temperature for the loss at the DUT output is set to Use Reference temperature.

[SENSe]:NOISe:CORRection:TCOLd:ADUT[:VALue]

Sets or queries the temperature value (K) for the loss at the DUT output in the noise displays.

Conditions

Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group

Sense commands

Syntax

```
[SENSe]:NOISe:CORRection:TCOLd:ADUT[:VALue] <NR3>
[SENSe]:NOISe:CORRection:TCOLd:ADUT[:VALue]?
```

Arguments

<NR3> = floating point value with an exponent.

Range: 0 K to 500 K

Returns

<NR2> = floating point value without an exponent, which is the temperature in Kelvin.

Examples

`SENS:NOIS:CORR:TCOL:ADUT:VAL?` might return 100.0000000000, indicating that the temperature for the loss at the DUT output is 100 K.

[SENSe]:NOISe:CORRection:TCOLd:BDUT:UREFerence

Sets or queries the analyzer to use the reference temperature (K) for the loss at the DUT input in the noise displays.

Conditions Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group Sense commands

Syntax

```
[SENSe]:NOISe:CORRection:TCOLd:BDUT:UREFerence { OFF | ON
| 1 | 0 }
[SENSe]:NOISe:CORRection:TCOLd:BDUT:UREFerence?
```

Arguments OFF or 0 disables the Use Ref temperature feature for the Loss at DUT Input.

ON or 1 enables the Use Ref temperature feature for the Loss at DUT Input.

Returns 0 means that the Use Ref temperature feature for the Loss at DUT Input is disabled.

1 means that the Use Ref temperature feature for the Loss at DUT Input is enabled.

Examples `SENS:NOIS:CORR:TCOL:BDUT:UREF?` might return 1, indicating that the temperature for the loss at the DUT input is set to Use Reference temperature.

[SENSe]:NOISe:CORRection:TCOLd:BDUT[:VALue]

Sets or queries the temperature value (K) for the loss at the DUT input in the noise displays.

Conditions Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group Sense commands

Syntax

```
[SENSe]:NOISe:CORRection:TCOLd:BDUT[:VALue] <NR3>
[SENSe]:NOISe:CORRection:TCOLd:BDUT[:VALue]?
```

Arguments <NR3> = floating point value with an exponent.

Range: 0 K to 500 K

Returns <NR2> = floating point value without an exponent, which is the temperature in Kelvin.

Examples SENS:NOIS:CORR:TCOL:BDUT:VAL 290 sets the temperature for the loss at the DUT input to 290 K.

[SENSe]:NOISe:CORRection:TCOLd[:REFerence]

Sets or queries the reference temperature value (K) in the noise display.

Conditions Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group Sense commands

Syntax
```
[SENSe]:NOISe:CORRection:TCOLd[:REFerence] <NR3>
[SENSe]:NOISe:CORRection:TCOLd[:REFerence]?
```

Arguments <NR3> = floating point value with an exponent.

Range: 0 K to 500 K

Returns <NR2> = floating point value without an exponent, which is the reference temperature in Kelvin.

Examples SENS:NOIS:CORR:TCOL:REF 200 sets the reference temperature to 200 K.

[SENSe]:NOISe:FIGure:POINTs:COUNt (No Query Form)

Sets the number of sample points for the Noise Figure display.

Conditions Measurement views: Noise Figure

Group Sense commands

Syntax
```
[SENSe]:NOISe:FIGure:POINTs:COUNt { P801 | P2401 | P4001 |
P8001 | P10401 }
```

Arguments

`P801` sets the number of sample points to 801.

`P2401` sets the number of sample points to 2401.

`P4001` sets the number of sample points to 4001.

`P8001` sets the number of sample points to 8001.

`P10401` sets the number of sample points to 10401.

Examples

`SENSE:NOISE:FIGURE:POINTS:COUNT P801` sets the number of sample points to 801.

[SENSe]:NOISe:FREQuency:CENTer

Sets or queries the center frequency value (Hz).

Conditions

Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group

Sense commands

Syntax

```
[SENSe]:NOISe:FREQuency:CENTer <NRf>
[SENSe]:NOISe:FREQuency:CENTer?
```

Arguments

`<NRf>` is the center frequency value in Hz.

Range: 0 Hz to 6.2 GHz

Returns

`<NR3>` = floating point value with an exponent, which is the center frequency value.

Examples

`[SENSE]:NOISE:FREQUENCY:CENTER 5.0 GHz` sets the center frequency in the active noise display to 5.0000 GHz.

`[SENSE]:NOISE:FREQUENCY:CENTER?` might return 6.2000000000E+9, indicating that the center frequency in the active noise display is 6.2000 GHz.

[SENSe]:NOISe:FREQuency:MODE

Sets or queries the frequency mode in the noise display.

Conditions Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group Sense commands

Syntax
```
[SENSe]:NOISe:FREQuency:MODE { TABLe | SINGle | SWEPt }
[SENSe]:NOISe:FREQuency:MODE?
```

Arguments

`TABLe`: sets the frequency mode to Frequency Table.

`SINGle`: sets the frequency mode to Single Frequency.

`SWEPt`: sets the frequency mode to Sweep Frequency.

Returns

`TABL`: means the frequency mode is set to Frequency Table.

`SING`: means the frequency mode is set to Single Frequency.

`SWEP`: means the frequency mode is set to Sweep Frequency.

Examples `SENS:NOIS:FREQ:MODE TABL` sets the frequency mode to Frequency Table in the noise display.

[SENSe]:NOISe:FREQuency:SPAN

Sets or queries the frequency Span value (Hz).

Conditions Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group Sense commands

Syntax
```
[SENSe]:NOISe:FREQuency:SPAN <NRf>
[SENSe]:NOISe:FREQuency:SPAN?
```

Arguments

`<NRf>` is the frequency Span value in Hz.

Range: 100 Hz to 6.2 GHz

Returns `<NR3>` = floating point value with an exponent, which is the frequency Span value.

Examples

`[SENSE]:NOISE:FREQUENCY:SPAN 3.0 GHz` sets the frequency Span in the active noise display to 3.0000 GHz.

`[SENSE]:NOISE:FREQUENCY:SPAN?` might return 100.000000000E+3, indicating that the frequency Span in the active noise display is 100 kHz.

[SENSe]:NOISe:FREQuency:STARt

Sets or queries the Start Frequency value.

Conditions

Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group

Sense commands

Syntax

```
[SENSe]:NOISe:FREQuency:STARt <NRf>
[SENSe]:NOISe:FREQuency:STARt?
```

Arguments

<NRf> is the Start Frequency value.

Returns

<NR3> = floating point value with an exponent, which is the Start Frequency value.

Examples

`[SENSE]:NOISE:FREQUENCY:START 1.5 GHz` sets the Start Frequency in the active noise display to 1.5000 GHz.

`[SENSE]:NOISE:FREQUENCY:START?` might return 1.5000000000E+9, indicating that the Start Frequency in the active noise display is 1.5000 GHz.

[SENSe]:NOISe:FREQuency:STEPs

Sets or queries the frequency Steps value.

Conditions

Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group

Sense commands

Syntax

```
[SENSe]:NOISe:FREQuency:STEPs <NRf>
[SENSe]:NOISe:FREQuency:STEPs?
```

Arguments <NRf> is the Steps value.

Range: 2 to 999

Returns <NRf> is the Steps value.

Examples [SENSE]:NOISE:FREQUENCY:STEPS 11 sets the frequency Steps value in the active noise display to 11.

[SENSe]:NOISe:FREQuency:STOP

Sets or queries the Stop Frequency value.

Conditions Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group Sense commands

Syntax
```
[SENSe]:NOISe:FREQuency:STOP <NRf>
[SENSe]:NOISe:FREQuency:STOP?
```

Arguments <NRf> is the Stop Frequency value.

Returns <NR3> = floating point value with an exponent, which is the Stop Frequency value.

Examples [SENSE]:NOISE:FREQUENCY:STOP 1.5 GHz sets the Stop Frequency in the active noise display to 1.5000 GHz.

[SENSE]:NOISE:FREQUENCY:STOP? might return 1.5000000000E+9, indicating that the Stop Frequency in the active noise display is 1.5000 GHz.

[SENSe]:NOISe:FREQuency:TABLe:DATA? (Query Only)

Queries the data in the Frequency Table for the noise display.

Conditions Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group Sense commands

Syntax `[SENSe]:NOISe:FREQuency:TABLe:DATA?`

Returns <string> which contains each frequency value (Hz) in the frequency table for the noise display.

Examples `SENSE:NOISE:FREQUENCY:TABLE:DATA?` might return "1500000000,1505000005,1510000010,1515000015,1520000020,1525000025,1530000030 1545000045,1550000050," indicating each frequency in the Frequency Table.

[SENSe]:NOISe:FREQuency:TABLe:NEW (No Query Form)

Creates a new Frequency Table with the specified values for the noise display.

Conditions Measurement views: Noise Figure, Gain, Noise Temperature, Y Factor, Noise Table

Group Sense commands

Syntax `[SENSe]:NOISe:FREQuency:TABLe:NEW <string>`

Arguments <string> = "<NR3>,<NR3>,<NR3>,..."

Examples `SENSE:NOISE:FREQUENCY:TABLE:NEW "1.2e9,1.5e9,1.452e9"` sets 3 step values in a new Frequency Table to 1.200 GHz, 1.350 GHz, and 1.500 GHz.

[SENSe]:OBWidth:AVERage

Sets or queries whether to enable averaging in the Occupied Bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Sense commands

Syntax
```
[SENSe]:OBWidth:AVERage { OFF | ON | 0 | 1 }
[SENSe]:OBWidth:AVERage?
```

Arguments `OFF` or `0` disables averaging.

`ON` or `1` enables averaging.

Examples `SENSE:OBWIDTH:AVERAGE ON` enables averaging.

[SENSe]:OBWidth:AVERage:COUNt

Sets or queries the number of measurements for averaging in the Occupied Bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Sense commands

Syntax
```
[SENSe]:OBWidth:AVERage:COUNt <number>
[SENSe]:OBWidth:AVERage:COUNt?
```

Arguments `<number> ::` `<NR1>` specifies the average count. Range: 2 to 10000.

Examples `SENSE:OBWIDTH:AVERAGE:COUNT 64` sets the average count to 64.

[SENSe]:OBWidth:{BANDwidth|BWIDth}:MEASurement

Sets or queries the measurement bandwidth to determine the total power in the Occupied Bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Sense commands

Syntax
```
[SENSe]:OBWidth:{BANDwidth|BWIDth}:MEASurement <value>
[SENSe]:OBWidth:{BANDwidth|BWIDth}:MEASurement?
```

Arguments `<value> ::` `<NRf>` specifies the measurement bandwidth. Range: 100 Hz to 109 MHz.

Examples `SENSE:OBWIDTH:BANDWIDTH:MEASUREMENT 10MHz` sets the measurement bandwidth to 10 MHz.

[SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution]

Sets or queries the resolution bandwidth (RBW). Programming a specified RBW sets [SENSe]:OBWidth{BANDwidth|BWIDth}[:RESolution]:AUTO OFF.

Conditions Measurement views: Occupied Bandwidth

Group Sense commands

Syntax

```
[SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution] <value>
[SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution]?
```

Related Commands [SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution]:AUTO

Arguments `<value> ::  <NRf>` specifies the RBW. Range: 100 Hz to 5 MHz.

Examples `SENSE:OBWIDTH:BANDWIDTH:RESOLUTION 200kHz` sets the RBW to 200 kHz.

[SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution]:ACTual? (Query Only)

Queries the actual resolution bandwidth (RBW) in the Occupied Bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Sense commands

Syntax

```
[SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution]:ACTual?
```

Arguments None

Returns `<NRf>` The actual RBW in Hz.

Examples `SENSE:OBWIDTH:BANDWIDTH:RESOLUTION:ACTUAL?` might return `299.624E+3`, indicating that the actual RBW is 299.624 kHz.

[SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution]:AUTO

Determines whether to set the resolution bandwidth (RBW) automatically or manually in the Occupied Bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Sense commands

Syntax
```
[SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution]:AUTO { OFF | ON | 0 | 1 }
[SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution]:AUTO?
```

Arguments `OFF` or `0` specifies that the RBW is set manually using the [SENSe]:OBWidth:{BANDwidth|BWIDth}[:RESolution] command.

`ON` or `1` specifies that the RBW is set automatically.

Examples `SENSE:OBWIDTH:BANDWIDTH:AUTO ON` sets the RBW automatically.

[SENSe]:OBWidth:{BANDwidth|BWIDth}:VIDeo

Sets or queries the video bandwidth (VBW). Programming a specified VBW sets [SENSe]:OBWidth{BANDwidth|BWIDth}:VIDeo:STATe OFF.

Conditions Measurement views: Occupied Bandwidth

Group Sense commands

Syntax
```
[SENSe]:OBWidth:{BANDwidth|BWIDth}:VIDeo <value>
[SENSe]:OBWidth:{BANDwidth|BWIDth}:VIDeo?
```

Arguments `<value> :: <NRf>` specifies the VBW.
Range: Current RBW/10^4 (1 Hz minimum) to Current RBW.

Examples `SENSE:OBWIDTH:BANDWIDTH:VIDEO 200kHz` sets the VBW to 200 kHz.

[SENSe]:OBWidth:{BANDwidth|BWIDth}:VIDeo:STATe

Determines whether to enable the video bandwidth (VBW) in the Occupied Bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Sense commands

Syntax

```
[SENSe]:OBWidth:{BANDwidth|BWIDth}:VIDeo:STATe { OFF | ON
| 0 | 1 }
[SENSe]:OBWidth:{BANDwidth|BWIDth}:VIDeo:STATe?
```

Arguments `OFF` or `0` disables the VBW.

`ON` or `1` enables the VBW.

Examples `SENSE:OBWIDTH:BANDWIDTH:VIDEO:STATE ON` enables the VBW.

[SENSe]:OBWidth:BOBW:XDBM:MEASurement:DIREction

Sets or queries the measurement direction in the Bluetooth window.

Conditions Measurement views:Bluetooth 20 dB Bandwidth

Group Sense commands

Syntax

```
[SENSe]:OBWidth:BOBW:XDBM:MEASurement:DIREction {IN|OUT}
[SENSe]:OBWidth:BOBW:XDBM:MEASurement:DIREction?
```

Arguments IN specifies the inward direction

OUT specifies the outward direction.

Examples `[SENSE]:OBWIDTH:BOBW:XDBM:MEASUREMENT:DIRECTION?` might return IN indicating the measurement direction is set to IN.

[SENSe]:OBWidth:BOBW:XDBM:RANGe

Sets or queries the measurement range in the Bluetooth window.

Conditions Measurement views:Bluetooth 20 dB Bandwidth

The x dBm option should be selected. If the x dB option is selected, the Range is always set to BOTH.

Group Sense commands

Syntax

```
[SENSe]:OBWidth:BOBW:XDBM:RANGe {BOTH|LOW|HIGH}
[SENSe]:OBWidth:BOBW:XDBM:RANGe?
```

Arguments BOTH specifies both the high and low ranges.

LOW specifies the lower range.

HIGH specifies the higher range.

Examples `[SENSE]:OBWIDTH:BOBW:XDBM:RANGE` LOW Sets the measurement range to the lower range.

[SENSe]:OBWidth:CLEar:RESults (No Query Form)

Restarts the average trace. This command is valid when [SENSe]:OBWidth:AVERage is set to ON.

Conditions Measurement views: Occupied Bandwidth

Group Sense commands

Syntax

```
[SENSe]:OBWidth:CLEar:RESults
```

Arguments None

Examples `SENSE:OBWIDTH:CLEAR:RESULTS` restarts the average trace.

[SENSe]:OBWidth:FREQuency:CENTer

Sets or queries the center frequency in the Occupied Bandwidth measurement.

Conditions Measurement views: Occupied Bandwidth

Group Sense commands

Syntax

```
[SENSe]:OBWidth:FREQuency:CENTer <value>
[SENSe]:OBWidth:FREQuency:CENTer?
```

Arguments `<value> ::= <NRf>` specifies the center frequency.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `SENSE:OBWIDTH:FREQUENCY:CENTER 7.5GHz` sets the center frequency to 7.5 GHz.

[SENSe]:OBWidth:FREQuency:STEP

Sets or queries the frequency step size in the Occupied Bandwidth measurement. Programming a specified step size sets [SENSe]:OBWidth:FREQuency:STEP AUTO OFF.

Conditions Measurement views: Occupied Bandwidth

Group Sense commands

Syntax

```
[SENSe]:OBWidth:FREQuency:STEP <value>
[SENSe]:OBWidth:FREQuency:STEP?
```

Related Commands [SENSe]:OBWidth:FREQuency:STEP:AUTO

Arguments

<value> :: <NRf> specifies the frequency step size.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

SENSE:OBWIDTH:FREQUENCY:STEP 1kHz sets the frequency step size to 1 kHz.

[SENSe]:OBWidth:FREQuency:STEP:AUTO

Determines whether to set the frequency step size automatically or manually in the Occupied Bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Sense commands

Syntax

```
[SENSe]:OBWidth:FREQuency:STEP:AUTO { OFF | ON | 0 | 1 }
[SENSe]:OBWidth:FREQuency:STEP:AUTO?
```

Arguments

OFF or 0 specifies that the frequency step size is set manually using the [SENSe]:OBWidth:FREQuency:STEP command.

ON or 1 specifies that the frequency step size is set automatically.

Examples

SENSE:OBWIDTH:FREQUENCY:STEP:AUTO ON specifies that the frequency step size is set automatically.

[SENSe]:OBWidth:PERCent

Sets or queries the occupied bandwidth percent power (power ratio of the occupied bandwidth to the measurement bandwidth).

Conditions

Measurement views: Occupied Bandwidth

Group

Sense commands

Syntax

```
[SENSe]:OBWidth:PERCent <value>
[SENSe]:OBWidth:PERCent?
```

Arguments

<value> :: <NRf> specifies the occupied bandwidth percent power. Range: 50 to 99.9%.

Examples

SENSE:OBWIDTH:PERCENT 98 sets the occupied bandwidth percent power to 98%.

[SENSe]:OBWidth:XDBLevel

Sets or queries the x dB level (how far down from the peak level the bandwidth is measured) in the x dB bandwidth measurement.

Conditions

Measurement views: Occupied Bandwidth

Group

Sense commands

Syntax

```
[SENSe]:OBWidth:XDBLevel <value>
[SENSe]:OBWidth:XDBLevel?
```

Arguments

<value> :: <NRf> specifies the x dB level. Range: -80 to -1 dB.

Examples

SENSE:OBWIDTH:XDBLEVEL -10 sets the x dB level to -10 dB.

[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth]

Sets or queries the channel bandwidth in the OFDM view.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth] <value>
[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth]?
```

Arguments

<value>::<NRf> specifies the channel bandwidth.

Examples `[SENSE]:OFDM:CHANNEL[:BANDWIDTH|BWIDTH] 20E+6` sets the channel bandwidth to 20 MHz.

[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth]:SRATe? (Query Only)

Returns the FFT sample rate for the OFDM view.

Conditions Measurement views: OFDM

Group Sense commands

Syntax `[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth]:SRATe?`

Arguments None

Returns <NRf> The sample rate in Hz.

Examples `[SENSE]:OFDM:CHANNEL[:BANDWIDTH|BWIDTH]:SRATE?` might return `20E+6` indicating the sample rate is 20 MHz.

[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth]:STATe

Determines whether the channel bandwidth is on or off in the OFDM view.

Conditions Measurement views: OFDM

Group Sense commands

Syntax
```
[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth]:STATe { OFF | ON | 0 | 1 }
[SENSe]:OFDM:CHANnel[:BANDwidth|BWIDth]:STATe?
```

Arguments `OFF` or `0` specifies that the channel bandwidth is off..

`ON` or `1` specifies that the channel bandwidth is on.

Examples `[SENSE]:OFDM:CHANNEL[:BANDWIDTH|BWIDTH]:STATE ON` sets the channel bandwidth setting to on.

[SENSe]:OFDM:CHANnel:ESTimation

Sets or queries the channel estimation in the OFDM view.

Conditions Measurement views: OFDM

Group Sense commands

Syntax

```
[SENSe]:OFDM:CHANnel:ESTimation { PREamble | PDATa }
[SENSe]:OFDM:CHANnel:ESTimation?
```

Arguments `PREamble` specifies the channel estimation is Preamble.

`PDATa` specifies the channel estimation is Preamble + Data.

Examples `[SENSE]:OFDM:CHANNEL:ESTIMATION PDAT` sets the channel estimation to Preamble + Data.

[SENSe]:OFDM:CONSte:DETermination

Sets or queries the constellation determination in the OFDM view.

Conditions Measurement views: OFDM

Group Sense commands

Syntax

```
[SENSe]:OFDM:CONSte:DETermination { AUTO | MANual }
[SENSe]:OFDM:CONSte:DETermination?
```

Arguments `AUTO` specifies the instrument automatically detects the constellation determination.

`MANual` allows you to select a manual constellation type.

Examples `[SENSE]:OFDM:CONSTE:DETERMINATION MAN` sets the constellation determination so you can select a manual constellation type using the [SENSe]:OFDM:CONSte:DETermination:MANual command.

[SENSe]:OFDM:CONSte:DETermination:MANual

Sets or queries a manual constellation type. The Constellation determination must be set to Manual using the [SENSe]:OFDM:CONSte:DETermination command.

Conditions Measurement views: OFDM

Group Sense commands

Syntax
```
[SENSe]:OFDM:CONSte:DETermination:MANual { BPSK | QAM16 | QAM64 | QPSK }
[SENSe]:OFDM:CONSte:DETermination:MANual?
```

Related Commands [SENSe]:OFDM:CONSte:DETermination

Arguments

`BPSK` specifies the BPSKconstellation type.

`QAM16` specifies the QAM16 constellation type.

`QAM64` specifies the QAM16 constellation type.

`QPSK` specifies the QPSK constellation type.

Examples `[SENSE]:OFDM:CONSTE:DETERMINATION:MANUAL QPSK` selects the QPSK constellation type.

[SENSe]:OFDM:FFT:LENGth? (Query Only)

Returns the FFT length for the OFDM view.

Conditions Measurement views: OFDM

Group Sense commands

Syntax
```
[SENSe]:OFDM:FFT:LENGth?
```

Arguments None

Returns <NR1> The FFT length.

Examples `[SENSE]:OFDM:FFT:LENGTH?` might return the FFT length as `64`.

[SENSe]:OFDM:GUARd:INTerval

Sets or queries the guard interval in the OFDM view.

Conditions Measurement views: OFDM

Group Sense commands

Syntax
```
[SENSe]:OFDM:GUARd:INTerval { FOURth | EIGHth | SIXTeenth
|THIRtysecond | USER }
```

Arguments `FOURth` specifies the guard interval is set to 1/4.

`EIGHth` specifies the guard interval is set to 1/8.

`SIXTeenth` specifies the guard interval is set to 1/16.

`THIRtysecond` specifies the guard interval is set to 1/32.

`USER` specifies a user-defined guard interval.

Examples `[SENSE]:OFDM:GUARD:INTERVAL THIR` sets the guard interval to 1/32.

[SENSe]:OFDM:GUARd:INTerval:USER

Sets or queries a user-defined guard interval. The Guard Interval must be set to USER.

Conditions Measurement views: OFDM

Group Sense commands

Syntax
```
[SENSe]:OFDM:GUARd:INTerval:USER <value>
[SENSe]:OFDM:GUARd:INTerval:USER?
```

Related Commands [SENSe]:OFDM:GUARd:INTerval

Arguments <value>::<NRf> specifies the guard interval.

Examples `[SENSE]:OFDM:GUARD:INTERVAL:USER 3.12` sets the guard interval to 3.12%.

[SENSe]:OFDM:PILot:TRACking:AMPLitude:STATe

Determines whether to select the Amplitude pilot tracking in the OFDM view.

Conditions Measurement views: OFDM

Group Sense commands

Syntax
```
[SENSe]:OFDM:PILot:TRACking:AMPLitude:STATe { OFF | ON | 0 | 1 }
[SENSe]:OFDM:PILot:TRACking:AMPLitude:STATe?
```

Arguments `OFF` or `0` specifies that the Amplitude pilot tracking is off..

`ON` or `1` specifies that the Amplitude pilot tracking is on.

Examples `[SENSE]:OFDM:PILOT:TRACKING:AMPLITUDE:STATE ON` turns the Amplitude pilot tracking on.

[SENSe]:OFDM:PILot:TRACking:PHASe:STATe

Determines whether to select the Phase pilot tracking in the OFDM view.

Conditions Measurement views: OFDM

Group Sense commands

Syntax
```
[SENSe]:OFDM:PILot:TRACking:PHASe:STATe { OFF | ON | 0 | 1 }
```

Arguments `OFF` or `0` specifies that the Phase pilot tracking is off..

`ON` or `1` specifies that the Phase pilot tracking is on.

Examples `[SENSE]:OFDM:PILOT:TRACKING:PHASE:STATE ON` turns the Phase pilot tracking on.

[SENSe]:OFDM:PILot:TRACking:TIMing:STATe

Determines whether to select the Timing pilot tracking in the OFDM view.

Conditions Measurement views: OFDM

Group Sense commands

Syntax `[SENSe]:OFDM:PILot:TRACking:TIMing:STATe { OFF | ON | 0 | 1 }`

Arguments `OFF` or `0` specifies that the Timing pilot tracking is off..

`ON` or `1` specifies that the Timing pilot tracking is on.

Examples `[SENSE]:OFDM:PILOT:TRACKING:TIMING:STATE ON` turns the Timing pilot tracking on.

[SENSe]:OFDM:RADix (No Query Form)

Sets the radix for the OFDM measurements. This command is effective in the symbol table.

Conditions Measurement views: OFDM

Group Sense commands

Syntax `[SENSe]:OFDM:RADix { BINary | HEX }`

Arguments

Returns `BINary` specifies the binary radix.

`HEX` specifies the hexadecimal radix.

Examples `[SENSE]:OFDM:RADIX` HEX sets the radix to hexadecimal for the OFDM symbol table.

[SENSe]:OFDM:SCARrier:SPACing

Sets or queries the subcarrier spacing in the OFDM view.

Conditions Measurement views: OFDM

Group Sense commands

Syntax
```
[SENSe]:OFDM:SCARrier:SPACing <value>
[SENSe]:OFDM:SCARrier:SPACing?
```

Arguments <value>::<NRf> specifies the subcarrier spacing.

Examples `[SENSE]:OFDM:SCARRIER:SPACING 312.5E+3` sets the subcarrier spacing to 312.5 MHz.

[SENSe]:OFDM:SCARrier:SPACing:STATe

Determines whether the subcarrier spacing is on or off in the OFDM view

Conditions Measurement views: OFDM

Group Sense commands

Syntax
```
[SENSe]:OFDM:SCARrier:SPACing:STATe { OFF | ON | 0 | 1 }
[SENSe]:OFDM:SCARrier:SPACing:STATe?
```

Arguments `OFF` or `0` specifies that the subcarrier spacing is off..

`ON` or `1` specifies that the subcarrier spacing is on.

Examples `[SENSE]:OFDM:SCARRIER:SPACING:STATE ON` sets the subcarrier spacing to on.

[SENSe]:OFDM:SCARriers

Sets or queries the Subcarriers to display in the OFDM view.

Conditions Measurement views: OFDM

Group Sense commands

Syntax
```
[SENSe]:OFDM:SCARriers { ALL | PILot | DATA | RANGe | SINGle }
[SENSe]:OFDM:SCARriers?
```

Arguments

`ALL` specifies that all subcarriers will be displayed.

`PILot` specifies that only the pilot subcarriers will be displayed.

`DATA` specifies that only the data subcarriers will be displayed.

`RANGe` specifies that the subcarriers within a specified range will be displayed.

`SINGle` specifies that only a single subcarrier will be displayed.

Examples `[SENSE]:OFDM:SCARRIERS ALL` sets the OFDM view to display all subcarriers.

[SENSe]:OFDM:SCARriers:RANGe:STARt

Sets or queries the subcarrier start range in the OFDM view.

Conditions Measurement views: OFDM

Group Sense commands

Syntax
```
[SENSe]:OFDM:SCARriers:RANGe:STARt <value>
[SENSe]:OFDM:SCARriers:RANGe:STARt?
```

Related Commands [SENSe]:OFDM:SCARriers:RANGe:STOP

Arguments <NR1>

Examples `[SENSE]:OFDM:SCARRIERS:RANGE:START −20` sets the subcarrier start range to –20.

[SENSe]:OFDM:SCARriers:RANGe:STOP

Sets or queries the subcarrier stop range in the OFDM view.

Conditions Measurement views: OFDM

Group Sense commands

Syntax
```
[SENSe]:OFDM:SCARriers:RANGe:STOP <value>
[SENSe]:OFDM:SCARriers:RANGe:STOP?
```

Related Commands [SENSe]:OFDM:SCARriers:RANGe:STARt

Arguments <NR1>

Examples `[SENSE]:OFDM:SCARRIERS:RANGE:STOP 20` sets the subcarrier stop range to 20.

[SENSe]:OFDM:SCARriers:SINGle:INDex

Sets or queries the index of a single subcarrier in the OFDM view.

Conditions Measurement views: OFDM

Group Sense commands

Syntax
```
[SENSe]:OFDM:SCARriers:SINGle:INDex <Value>
[SENSe]:OFDM:SCARriers:SINGle:INDex?
```

Arguments <NR1>

Examples `[SENSE]:OFDM:SCARRIERS:SINGLE:INDEX –10` sets the single subcarrier index to –10.

[SENSe]:OFDM:STANdard

Sets or queries the OFDM standard.

Conditions Measurement views: OFDM

Group Sense commands

Syntax
```
[SENSe]:OFDM:STANdard { S802_11 | S802_16 }
[SENSe]:OFDM:STANdard?
```

Arguments `S802_11` specifies the 802.11a/g/j OFDM standard.

`S802_16` specifies the 802.16 (2004) standard..

Examples `[SENSE]:OFDM:STANDARD S802_16` sets the OFDM standard to 802.16.

[SENSe]:OFDM:SWAP:IQ

Determines whether or not to exchange I and Q data before demodulating.

Conditions Measurement views: OFDM

Group Sense commands

Syntax
```
[SENSe]:OFDM:SWAP:IQ { OFF | ON | 0 | 1 }
[SENSe]:OFDM:SWAP:IQ?
```

Arguments `OFF` or `0` uses I and Q data as they are.

`ON` or `1` exchanges I and Q data.

Examples `[SENSE]:OFDM:SWAP:IQ ON` exchanges I and Q data before demodulating.

[SENSe]:OFDM:SYMBol:ANALysis:OFFSet

Sets or queries the OFDM symbol analysis offset.

Conditions Measurement views: OFDM

Group Sense commands

Syntax

```
[SENSe]:OFDM:SYMBol:ANALysis:OFFSet <value>
[SENSe]:OFDM:SYMBol:ANALysis:OFFSet?
```

Arguments

<value> specifies the symbol analysis offset as a percentage.

Range 0 to –100%.

Examples

`[SENSE]:OFDM:SYMBOL:ANALYSIS:OFFSET –50` sets the symbol analysis offset to –50%.

[SENSe]:OFDM:SYMBols

Sets or queries the symbols to display in the OFDM view.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:SYMBols { ALL | RANGe | SINGle }
[SENSe]:OFDM:SYMBols?
```

Arguments

`ALL` specifies that all symbols will be displayed.

`RANGe` specifies that a range of symbols will be displayed.

`SINGle` specifies that single symbol will be displayed.

Examples

`[SENSE]:OFDM:SYMBOLS ALL` specifies that all symbols will be displayed in the OFDM views.

[SENSe]:OFDM:SYMBols:MAX

Sets or queries the maximum number of symbols to analyze in the OFDM views.

Conditions

Measurement views: OFDM

Group

Sense commands

Syntax

```
[SENSe]:OFDM:SYMBols:MAX <value>
[SENSe]:OFDM:SYMBols:MAX?
```

Arguments <NR1>

Examples `[SENSE]:OFDM:SYMBOLS:MAX 50` specifies a maximum of 50 symbols to analyze in the OFDM views.

[SENSe]:OFDM:SYMBols:MAX:STATe

Enables or disables the maximum number of symbols to analyze in the OFDM view.

Conditions Measurement views: OFDM

Group Sense commands

Syntax

```
[SENSe]:OFDM:SYMBols:MAX:STATe { OFF | ON | 0 | 1 }
[SENSe]:OFDM:SYMBols:MAX:STATe?
```

Related Commands [SENSe]:OFDM:SYMBols:MAX

Arguments `OFF` or `0` disables the maximum number of symbols to analyze.

`ON` or `1` enables the maximum number of symbols to analyze.

Examples `[SENSE]:OFDM:SYMBOLS:MAX:STATE` ON enables the number of symbols to analyze in the OFDM view. The number is specified by the [SENSe]:OFDM:SYMBols:MAX command.

[SENSe]:OFDM:SYMBols:RANGe:COUNt

Sets or queries the number of symbols to display in the OFDM view when displaying a range of symbols.

Conditions Measurement views: OFDM

Group Sense commands

Syntax

```
[SENSe]:OFDM:SYMBols:RANGe:COUNt<value>
[SENSe]:OFDM:SYMBols:RANGe:COUNt?
```

Arguments <NR1>

Examples `[SENSE]:OFDM:SYMBOLS:RANGE:COUNT 25` sets the number of symbols to display to 25.

[SENSe]:OFDM:SYMBols:RANGe:STARt

Sets or queries the symbols start range in the OFDM view when displaying a range of symbols.

Conditions Measurement views: OFDM

Group Sense commands

Syntax

```
[SENSe]:OFDM:SYMBols:RANGe:STARt <value>
[SENSe]:OFDM:SYMBols:RANGe:STARt?
```

Arguments <NR1>

Examples `[SENSE]:OFDM:SYMBOLS:RANGE:START 0` sets the start range to 0.

[SENSe]:OFDM:SYMBols:SINGle:INDex

Sets or queries the symbol index in the OFDM view when displaying a single symbol.

Conditions Measurement views: OFDM

Group Sense commands

Syntax

```
[SENSe]:OFDM:SYMBols:SINGle:INDex <value>
[SENSe]:OFDM:SYMBols:SINGle:INDex?
```

Arguments <NR1>

Examples `[SENSE]:OFDM:SYMBOLS:SINGLE:INDEX 0` sets the index to 0.

[SENSe]:OFDM:UNIT:FREQuency (No Query Form)

Specifies the frequency units for the OFDM measurement.

Conditions Measurement views: OFDM

Group Sense commands

Syntax `[SENSe]:OFDM:UNIT:FREQuency { FREQuency | SCARrier }`

Arguments FREQuency sets the frequency units to frequency.

SCARrier sets the frequency units to subcarriers.

Examples `[SENSE]:OFDM:UNIT:FREQUENCY FREQ` sets the frequency units to frequency.

[SENSe]:OFDM:UNIT:TIME (No Query Form)

Specifies the time units for the OFDM measurement.

Conditions Measurement views: OFDM

Group Sense commands

Syntax `[SENSe]:OFDM:UNIT:TIME { SEConds | SYMBols }`

Arguments SEConds sets the time units to seconds.

SYMBols sets the time units to symbols.

Examples `[SENSE]:OFDM:UNIT:TIME SEC` sets the time units to seconds.

[SENSe]:P25:ANALysis:OFFSet

Sets or queries the P25 Analysis Offset value.

Conditions Measurement view: Any P25 measurement display

Group Sense commands

Syntax
```
[SENSe]:P25:ANALysis:OFFSet <value>
[SENSe]:P25:ANALysis:OFFSet?
```

Related Commands [SENSe]:P25:ANALysis:OFFSet:AUTO

Arguments <value> specifies the symbol analysis offset value.

Returns <NR2> = a floating point value without an exponent, which is the analysis offset value.

Examples `[SENSE]:P25:ANALYSIS:OFFSET?` might return -67.0000000000, indicating that the Analysis Offset is -67.00 Symbol.

`[SENSE]:P25:ANALYSIS:OFFSET-57` sets the Analysis Offset to -57.000 Symbol.

[SENSe]:P25:ANALysis:OFFSet:AUTO

Sets or queries if the P25 Analysis Offset is set to Auto or if Auto is off.

Conditions Measurement view: Any P25 measurement display

Group Sense commands

Syntax
```
[SENSe]:P25:ANALysis:OFFSet:AUTO <value>
[SENSe]:P25:ANALysis:OFFSet:AUTO?
```

Related Commands [SENSe]:P25:ANALysis:OFFSet

Arguments `OFF` or `0` sets Analysis Offset to Auto off.

`ON` or `1` sets Analysis Offset to Auto on.

Returns `0` means Analysis Offset is set to Auto off.

1 means Analysis Offset is set to Auto on.

Examples

[SENSE]:P25:ANALYSIS:OFFSET:AUTO? might return -67.0000000000, indicating that the Analysis Offset is -67.00 Symbol.

[SENSE]:P25:ANALYSIS:OFFSET:AUTO−57 sets the Analysis Offset to -57.000 Symbol.

[SENSe]:P25([:BANDwidth]|:BWIDth):MODE

Sets or queries the Measurement BW (bandwidth) (frequency span) mode as Auto, Manual, or Link to Span.

Conditions

Measurement view: Any P25 measurement display

Group

Sense commands

Syntax

```
[SENSe]:P25([:BANDwidth]|:BWIDth):MODE { MANual | AUTO | LSP }
[SENSe]:P25([:BANDwidth]|:BWIDth):MODE?
```

Arguments

MANual sets the analyzer to allow for manual selection of the measurement bandwidth. A typical use is setting a narrow measurement bandwidth on a small range of frequencies to examine one signal while rejecting the others that are present in a Spectrum graph.

AUTO sets analyzer to automatically select a measurement bandwidth. The measurement picks a bandwidth based on other parameter settings (such as symbol rate and modulation type).

LSP sets the Measurement BW to Link to Span. This allows you to use the Spectrum display to tune the frequency, view the signal, and then use the Span control to set the measurement bandwidth. This mode emulates legacy instruments.

Returns

MAN means the Measurement BW is set to Manual.

AUTO means the Measurement BW is set to Automatic.

LSP means the Measurement BW is set to Link to Span.

Examples

SENS:P25:BAND:MODE? might return MAN, indicating the Measurement BW mode is set to AUTO.

[SENSe]:P25([:BANDwidth]|:BWIDth):TINTerval

Sets or queries the Measurement BW (bandwidth) (frequency span). Setting the Measurement BW changes this setting from Auto to Manual if it is set to Auto before you use the command. A query will not change that setting.

Conditions Measurement view: Any P25 measurement display

Group Sense commands

Syntax

```
[SENSe]:P25([:BANDwidth]|:BWIDth):TINTerval <NRf>
[SENSe]:P25([:BANDwidth]|:BWIDth):TINTerval?
```

Arguments <NRf> specifies the measurement bandwidth in kHz.

Returns <NR3> = a floating point value with an exponent, which is the Measurement BW value in kHz.

Examples `SENS:P25:BAND:INT?` might return 153.6000000000E+3, which indicates the Measurement BW value is 153.6 kHz.

[SENSe]:P25([:BANDwidth]|:BWIDth):TINTerval:AUTO

Sets or queries the Measurement BW (bandwidth) (frequency span) mode as Auto or Manual.

Conditions Measurement view: Any P25 measurement display

Group Sense commands

Syntax

```
[SENSe]:P25([:BANDwidth]|:BWIDth):TINTerval:AUTO { 0 | 1 |
OFF | ON }
[SENSe]:P25([:BANDwidth]|:BWIDth):TINTerval:AUTO?
```

Arguments `OFF` or `0` sets Measurement BW to Auto.

`ON` or `1` sets Measurement BW to Manual.

Returns

`0` means Measurement BW is set to Auto.

`1` means Measurement BW is set to Manual.

Examples

`SENS:P25:BAND:INT:AUTO?` might return 1, indicating the Measurement BW is set to Auto.

[SENSe]:P25:FILTer:MEASurement

Sets or queries the P25 measurements Measurement Filter setting. The User Meas Filter selections are predefined user filters.

Conditions

Measurement view: Any P25 measurement display

Group

Sense commands

Syntax

```
[SENSe]:P25:FILTer:MEASurement { USER1 | USER2 | OFF | UOTH |
USER3 | RECT | GAUS | RRC | OFF | RCOS | HDQPSK_P25 | C4FM_P25 }
[SENSe]:P25:FILTer:MEASurement?
```

Arguments

OFF sets the filter to None.

USER1 sets the filter to User Meas Filter 1.

USER2 sets the filter to User Meas Filter 2.

USER3 sets the filter to User Meas Filter 3.

UOTH sets the filter to User other.

RECT sets the filter to Rectangular (only available for Phase 2 HDQPSK modulation type).

RRC sets the filter to Root Raised Cosine (only available for Phase 2 HDQPSK modulation type).

RCOS sets the filter to Raised Cosine (only available for Phase 2 HDQPSK modulation type).

GAUS sets the filter to Gaussian (only available for Phase 2 HDQPSK modulation type).

HDQPSK_P25 sets the filter to HDQPSK-P25 (only available for Phase 2 HDQPSK modulation type)

C4FM_P25 sets the filter to C4FM-P25 (only available for Phase 1 C4FM modulation type)

Returns

OFF means the filter is set to None.

USER1 means the filter is set to User Meas Filter 1.

USER2 means the filter is set to User Meas Filter 2.

USER3 means the filter is set to User Meas Filter 3.

UOTH means the filter is set to User other.

RECT means the filter is set to Rectangular (only available for Phase 2 HDQPSK modulation type).

RRC means the filter is set to Root Raised Cosine (only available for Phase 2 HDQPSK modulation type).

RCOS means the filter is set to Raised Cosine (only available for Phase 2 HDQPSK modulation type).

GAUS means the filter is set to Gaussian (only available for Phase 2 HDQPSK modulation type).

HDQPSK_P25 means the filter is set to HDQPSK-P25 (only available for Phase 2 HDQPSK modulation type)

C4FM_P25 means the filter is set to C4FM-P25 (only available for Phase 1 C4FM modulation type)

Examples

`[SENSE]:P25:FILTER:MEASUREMENT?` might return USER1, indicating that the Measurement Filter is set to User Meas Filter 1.

[SENSe]:P25:FILTer:REFerence

Sets or queries the P25 measurements Reference Filter setting. The User Meas Filter selections are predefined user filters.

Conditions

Measurement view: Any P25 measurement display

Group

Sense commands

Syntax

```
[SENSe]:P25:FILTer:REFerence { USER1 | USER2 | OFF | UOTH |
USER3 | RECT | GAUS | RRC | OFF | RCOS | HDQPSK_P25 | C4FM_P25 }
[SENSe]:P25:FILTer:REFerence?
```

Arguments

OFF sets the filter to None.

USER1 sets the filter to User Meas Filter 1.

USER2 sets the filter to User Meas Filter 2.

USER3 sets the filter to User Meas Filter 3.

UOTH sets the filter to User other.

RECT sets the filter to Rectangular (only available for Phase 2 HDQPSK modulation type).

RCOS sets the filter to Raised Cosine (only available for Phase 2 HDQPSK modulation type).

GAUS sets the filter to Gaussian (only available for Phase 2 HDQPSK modulation type).

Returns OFF means the filter is set to None.

USER1 means the filter is set to User Meas Filter 1.

USER2 means the filter is set to User Meas Filter 2.

USER3 means the filter is set to User Meas Filter 3.

UOTH means the filter is set to User other.

RECT means the filter is set to Rectangular (only available for Phase 2 HDQPSK modulation type).

RCOS means the filter is set to Raised Cosine (only available for Phase 2 HDQPSK modulation type).

GAUS means the filter is set to Gaussian (only available for Phase 2 HDQPSK modulation type).

Examples `[SENSE]:P25:FILTER:REFERENCE?` might return USER1, indicating that the Reference Filter is set to User Meas Filter 1.

[SENSe]:P25:FREQuency:ERRor

Sets or queries the P25 Frequency Offset (when Auto is off).

Conditions Measurement view: Any P25 measurement display

Frequency Offset set to manual (Auto off).

Group Sense commands

Syntax

```
[SENSe]:P25:FREQuency:ERRor <NR2>
[SENSe]:P25:FREQuency:ERRor?
```

Related Commands

Arguments <NR2> = a floating point value without an exponent, which is the fixed frequency offset in Hz.

Returns <NR2> = a floating point value without an exponent, which is the fixed frequency offset in Hz.

Examples `[SENSE]:P25:FREQUENCY:ERROR?` might return 749.9999390000, indicating that the frequency offset is 750.0 Hz.

`[SENSE]:P25:FREQUENCY:ERROR700.99` sets the frequency offset the 701.0 Hz.

[SENSe]:P25:FREQuency:ERRor:AUTO

Sets or queries if the P25 Frequency Error is set to Auto.

Conditions Measurement view: Any P25 measurement display

Group Sense commands

Syntax

```
[SENSe]:P25:FREQuency:ERRor:AUTO { 0 | 1 | OFF | ON }
[SENSe]:P25:FREQuency:ERRor:AUTO?
```

Related Commands

Arguments `OFF` or `0` means the Auto is off.

`ON` or `1` means the Auto is on.

Returns `0` means that Auto is off.

`1` means Auto is on.

Examples `[SENSE]:P25:FREQUENCY:ERROR:AUTO?` might return 1, indicating that the Frequency Error is set to Auto.

`[SENSE]:P25:FREQUENCY:ERROR:AUTO0` turns off Auto setting for Frequency Error.

[SENSe]:P25:MODUlation:CHANnel:([BANDwidth]|:BWIDth)]? (Query Only)

Queries the actual P25 Preset channel bandwidth.

Conditions Measurement view: Any P25 measurement display

Group Sense commands

Syntax `[SENSe]:P25:MODUlation:CHANnel:([BANDwidth]|:BWIDth)]?`

Returns Bandwidth in kHz, using '_" to denote ".". For example, 12_5 means 12.5 kHz.

Examples `SENS:P25:MODU:CHAN:BAND?` will return 12_5, indicating that the channel bandwidth is 12.5 kHz.

[SENSe]:P25:MODUlation:STANdard

Sets or queries the P25 modulation standard.

Conditions Measurement view: Any P25 measurement display

Group Sense commands

Syntax
```
[SENSe]:P25:MODUlation:STANdard { PHASE1 | PHASE2 }
[SENSe]:P25:MODUlation:STANdard?
```

Related Commands SENSe:P25:MODUlation:TYPE

Arguments PHASE1 is the Phase 1 standard.

PHASE2 is the Phase 2 standard.

Returns PHASE1 is the Phase 1 standard.

PHASE2 is the Phase 2 standard.

Examples `[SENSE]:P25:MODULATION:STANDARD?` might return PHASE2, indicating that the P25 modulation standard is set to Phase 2.

SENSe:P25:MODUlation:TYPE

Sets or queries the P25 modulation type.

Conditions Measurement view: Any P25 measurement display

Group Sense commands

Syntax

```
SENSe:P25:MODUlation:TYPE { C4FM | HCPM | HDQPSK }
SENSe:P25:MODUlation:TYPE?
```

Related Commands [SENSe]:P25:MODUlation:STANdard

Arguments C4FM: this is a Phase 1 modulation standard type. Modulation Phase must be set to PHASE1.

HCPM: this is a Phase 2 modulation standard type. Modulation Phase must be set to PHASE2.

HDQPSK: this is a Phase 2 modulation standard type. Modulation Phase must be set to PHASE2.

Returns C4FM: this is a Phase 1 modulation standard type.

HCPM: this is a Phase 2 modulation standard type.

HDQPSK: this is a Phase 2 modulation standard type.

Examples `SENSE:P25:MODULATION:TYPE?` might return HDQPSK, indicating that the P25 modulation type is set to HDQPSK, which is a Phase 2 modulation standard.

[SENSe]:P25:SRATe

Sets or queries the Symbol Rate value for the P25 measurements.

Conditions Measurement view: Any P25 measurement display

Group Sense commands

Syntax

```
[SENSe]:P25:SRATe <value>
[SENSe]:P25:SRATe?
```

Arguments <value> is the symbol rate value in kHz.

Returns <NR3> = a floating point value with an exponent, which is the symbol rate value in kHz.

Examples `[SENSE]:P25:SRATE?` might return 6.0000000000E+3, indicating that the symbol rate is 6 kHz.

[SENSe]:P25:SUMMary:CLEar:RESults (No Query Form)

Clears the results of the P25 Summary display.

Conditions Measurement view: P25 Summary

Group Sense commands

Syntax `[SENSe]:P25:SUMMary:CLEar:RESults`

Arguments None

Examples `[SENSE]:P25:SUMMARY:CLEAR:RESULTS` clears the results in the P25 Summary display.

[SENSe]:P25:SUMMary:LIMIt:FA

Sets or queries for the On/Off state of limit comparison and the upper limit value for Operating Frequency Accuracy. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions Measurement view: P25 Summary

Group Sense commands

Syntax
```
[SENSe]:P25:SUMMary:LIMIt:FA <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FA?
```

Arguments <NR1>=0 to deselect the measurement and 1 to select the measurement.

<NR2>=0 to 4

Returns <NR1>,<NR2>

Examples `[SENSE]:P25:SUMMARY:LIMIT:FA?` might return "1,2" <1> indicates measurement is enabled (turned on) for limit comparison, and the upper limit for Operating Frequency Accuracy is <2>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:AVERage:TERRor:ONE

Sets or queries for the On/Off state of limit comparison and the upper limit value for Average t_error_1. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions Measurement view: P25 Summary

Group Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:AVERage:
TERRor:ONE <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:AVERage:
TERRor:ONE?
```

Arguments <NR1>=0 to deselect the measurement and 1 to select the measurement.

<NR2>=0 to 25

Returns <NR1>,<NR3>

Examples `[SENSE]:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:AVERAGE:TERROR:ONE?` might return "0,25.0000000000E-6" <0> indicating measurement is disabled (turned off) for limit comparison, and the upper limit for Average t_error_1 is <25.0000000000E-6>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:AVERage:TERRor:ZERO

Sets or queries for the On/Off state of limit comparison and the upper limit value for Average t_error_0. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions Measurement view: P25 Summary

Group Sense commands

Syntax
```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:AVERage:
TERRor:ZERO <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:AVERage:
TERRor:ZERO?
```

Arguments <NR1>=0 to deselect the measurement and 1 to select the measurement.

<NR2>=0 to 25

Returns <NR1>,<NR3>

Examples `[SENSE]:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:AVERAGE:TERROR:ZERO?` might return "0,25.0000000000E-6" <0> indicating measurement is disabled (turned off) for limit comparison, and the upper limit for Average t_error_0 is <25.0000000000E-6>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:LOGIc:CHANnel:OFF:SLOT

Sets or queries for the On/Off state of limit comparison and the upper limit value for HCPM Transmitter Logic channel offslot. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions Measurement view: P25 Summary

Group Sense commands

Syntax
```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:LOGIc:CHANnel:
OFF:SLOT <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:LOGIc:CHANnel:
OFF:SLOT?
```

Arguments <NR1>=0 to deselect the measurement and 1 to select the measurement.

<NR2>=0 to -57

Returns <NR1>,<NR2>

Examples `[SENSE]:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:LOGIC:CHANNEL:OFF:SLOT?` might return "0,-57" <0> indicates measurement is disabled (turned off) for limit comparison, and the upper limit for HCPM Transmitter Logic channel offslot is <-57>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:LOGIc:CHANnel:PEAK:ACPR

Sets or queries for the On/Off state of limit comparison and the upper limit value for HCPM Transmitter Logic channel Peak ACPR. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions Measurement view: P25 Summary

Group Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:LOGIc:CHANnel:PEAK:ACPR <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:LOGIc:CHANnel:PEAK:ACPR?
```

Arguments <NR1>=0 to deselect the measurement and 1 to select the measurement.

<NR2>=0 to 35

Returns <NR1>,<NR2>

Examples `[SENSE]:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:LOGIC:CHANNEL:PEAK:ACPR?` might return "0,35" <0> indicating measurement is disabled (turned off) for limit comparison, and the upper limit for HCPM Transmitter Logic channel Peak ACPR is <35>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:NEGAtive:LO

Sets or queries for the On/Off state of limit comparison and the lower limit value for Frequency Deviation, negative peak. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions Measurement view: P25 Summary

Group Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:NEGAtive:LO <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:NEGAtive:LO?
```

Arguments

<NR1>=0 to deselect the measurement and 1 to select the measurement.

<NR2>=0 to 2544

Returns

<NR1>,<NR2>

Examples

`[SENSE]:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PEAK:NEGATIVE:LO?` might return "1,2544" <1> indicates measurement is enabled (turned on) for limit comparison, and the lower limit for Frequency Deviation, negative peak, is <2544>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:NEGAtive:UP

Sets or queries for the On/Off state of limit comparison and the upper limit value for Frequency Deviation, negative peak. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:NEGAtive:UP <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PEAK:NEGAtive:UP?
```

Arguments

<NR1>=0 to deselect the measurement and 1 to select the measurement.

<NR2>=0 to 3111

Returns

<NR1>,<NR2>

Examples

`[SENSE]:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PEAK:NEGATIVE:UP?` might return "1,3111" <1> indicates measurement is enabled (turned on) for limit comparison, and the upper limit for Frequency Deviation, negative peak, is <3111>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:PEAK:POSItive:LO

Sets or queries for the On/Off state of limit comparison and the lower limit value for Frequency Deviation, positive peak. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions Measurement view: P25 Summary

Group Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:PEAK:POSItive:LO <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:PEAK:POSItive:LO?
```

Arguments <NR1>=0 to deselect the measurement and 1 to select the measurement.

<NR2>=0 to 2544

Returns <NR1>,<NR2>

Examples `[SENSE]:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PEAK:POSITIVE:LO?` might return “1,2544” <1> indicates measurement is enabled (turned on) for limit comparison, and the lower limit for Frequency Deviation, positive peak, is <2544>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:PEAK:POSItive:UP

Sets or queries for the On/Off state of limit comparison and the upper limit value for Frequency Deviation, positive peak. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions Measurement view: P25 Summary

Group Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:PEAK:POSItive:UP <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:PEAK:POSItive:UP?
```

Arguments <NR1>=0 to deselect the measurement and 1 to select the measurement.

<NR2>=0 to 3111 Hz

Returns <NR1>,<NR2>

Examples `[SENSE]:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PEAK:POSITIVE:UP?` might return "1,3111" <1> indicates measurement is enabled (turned on) for limit comparison, and the upper limit for Frequency Deviation, positive peak, is <3111>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PHASe[1/2]:F[1/2/3/4]:LO

Sets or queries for the On/Off state of limit comparison and the lower limit value for Frequency Deviation, f1/f2/f3/f4 for Phase1 or Phase2. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions Measurement view: P25 Summary

Group Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PHASe[1/2]:F[1/2/3/4]:LO <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PHASe[1/2]:F[1/2/3/4]:LO?
```

Arguments <NR1>=0 to deselect the measurement and 1 to select the measurement.

<NR2>=0 to 2544

Returns <NR1>,<NR2>

Examples `[SENSE]:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PHASE[1/2]:F[1/2/3/4]:LO?` might return "1,2544" <1> indicating measurement is enabled (turned on) for limit comparison, and the upper limit for Frequency Deviation is <2544>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:PHASe[1/2]:F[1/2/3/4]:UP

Sets or queries for the On/Off state of limit comparison and the upper limit value for Frequency Deviation, f1/f2/f3/f4 for Phase1 or Phase2. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions Measurement view: P25 Summary

Group Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:PHASe[1/2]:
F[1/2/3/4]:UP <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:PHASe[1/2]:
F[1/2/3/4]:UP?
```

Arguments <NR1>=0 to deselect the measurement and 1 to select the measurement.

<NR2>=0 to 3111

Returns <NR1>,<NR2>

Examples `[SENSE]:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PHASE[1/2]:F[1/2/3/4]:UP?` might return "1,3111" <1> indicating measurement is enabled (turned on) for limit comparison, and the upper limit for Frequency Deviation is <3111>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:PMAX:OFF

Sets or queries for the On/Off state of limit comparison and the upper limit value for Pmax-off. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions Measurement view: P25 Summary

Group Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:PMAX:OFF
<NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:PMAX:OFF?
```

Arguments <NR1>=0 to deselect the measurement and 1 to select the measurement.

<NR2>=0 to 4

Returns <NR1>,<NR2>

Examples `[SENSE]:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PMAX:OFF?` might return "0,4" <0> indicating measurement is disabled (turned off) for limit comparison, and the upper limit for Pmax-off is <4>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:PMAX:ON

Sets or queries for the On/Off state of limit comparison and the upper limit value for Pmax-on. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions Measurement view: P25 Summary

Group Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:PMAX:ON
<NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:PMAX:ON?
```

Arguments <NR1>=0 to deselect the measurement and 1 to select the measurement.

<NR2>=0 to 4

Returns <NR1>,<NR2>

Examples `[SENSE]:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PMAX:ON?` might return "0,4" <0> indicating measurement is disabled (turned off) for limit comparison, and the upper limit for Pmax-on is <4>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:PMAX:SS

Sets or queries for the On/Off state of limit comparison and the upper limit value for Pss-max. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions Measurement view: P25 Summary

Group Sense commands

Syntax
```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:SS
<NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMAX:SS?
```

Arguments <NR1>=0 to deselect the measurement and 1 to select the measurement.

<NR2>=0 to 1

Returns <NR1>,<NR2>

Examples `[SENSE]:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PMAX:SS?` might return "0,1" <0> indicating measurement is disabled (turned off) for limit comparison, and the upper limit for Pss-max is <1>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMIN:SS

Sets or queries for the On/Off state of limit comparison and the upper limit value for Pss-min. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions Measurement view: P25 Summary

Group Sense commands

Syntax
```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMIN:SS
<NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:PMIN:SS?
```

Arguments <NR1>=0 to deselect the measurement and 1 to select the measurement.

<NR2>=0 to -3

Returns <NR1>,<NR2>

Examples `[SENSE]:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:PMIN:SS?` might return "0,-3" <0> indicating measurement is disabled (turned off) for limit comparison, and the upper limit for Pss-min is <-3>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:TRANsmitter:AVERage:ATTK:ENCO

Sets or queries for the On/Off state of limit comparison and the upper limit value for Transmitter Average Attack encoder. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions Measurement view: P25 Summary

Group Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:TRANsmitter:
AVERage:ATTK:ENCOder <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:TRANsmitter:
AVERage:ATTK:ENCOder?
```

Arguments <NR1>=0 to deselect the measurement and 1 to select the measurement.

<NR2>=0 to 100

Returns <NR1>,<NR2>

Examples `[SENSE]:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:TRANSMITTER:AVERAGE:ATTK:ENCODER?` might return "0,100" <0> indicates measurement is disabled (turned off) for limit comparison, and the upper limit for Transmitter Average Attack encoder is <100>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEViation:TRANsmitter:AVERage:ATTK:ENCO

Sets or queries for the On/Off state of limit comparison and the upper limit value for Transmitter Average Attack encoder, busy/idle. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions Measurement view: P25 Summary

Group Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter:
AVERage:ATTK:ENCOder:BI <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter:
AVERage:ATTK:ENCOder:BI?
```

Arguments

<NR1>=0 to deselect the measurement and 1 to select the measurement.

<NR2>=0 to 30

Returns

<NR1>,<NR2>

Examples

`[SENSE]:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:TRANSMITTER:AVERAGE:ATTK:ENCODER:BI?` might return "0,30" <0> indicates measurement is disabled (turned off) for limit comparison, and the upper limit for Transmitter Average Attack encoder, busy/idle, is <30>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter:AVERage:ATTK:POWEr

Sets or queries for the On/Off state of limit comparison and the upper limit value for Transmitter Average Attack power. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions

Measurement view: P25 Summary

Group

Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter:
AVERage:ATTK:POWEr <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter:
AVERage:ATTK:POWEr?
```

Arguments

<NR1>=0 to deselect the measurement and 1 to select the measurement.

<NR2>=0 to 50

Returns

<NR1>,<NR2>

Examples

`[SENSE]:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:TRANSMITTER:AVERAGE:ATTK:POWER?` might return "0,50" <0> indicates measurement is disabled (turned off) for limit comparison, and the upper limit for Transmitter Average Attack power is <50>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter:AVERage:ATTK:POWE

Sets or queries for the On/Off state of limit comparison and the upper limit value for Transmitter Average Attack power, busy/idle. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions Measurement view: P25 Summary

Group Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter:
AVERage:ATTK:POWEr:BI <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter:
AVERage:ATTK:POWEr:BI?
```

Arguments <NR1>=0 to deselect the measurement and 1 to select the measurement.

<NR2>=0 to 30

Returns <NR1>,<NR2>

Examples `[SENSE]:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:TRANSMITTER:AVERAGE:ATTK:POWER:BI?` might return "0,30" <0> indicates measurement is disabled (turned off) for limit comparison, and the upper limit for Transmitter Average Attack power, busy/idle, is <30>.

[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter:AVERage:THRUput:DL

Sets or queries for the On/Off state of limit comparison and the upper limit value for Transmitter Average throughput delay. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions Measurement view: P25 Summary

Group Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter:
AVERage:THRUput:DLAY <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:FREQuency:DEVIation:TRANsmitter:
AVERage:THRUput:DLAY?
```

Arguments <NR1>=0 to deselect the measurement and 1 to select the measurement.

<NR2>=0 to 125

Returns <NR1>,<NR2>

Examples `[SENSE]:P25:SUMMARY:LIMIT:FREQUENCY:DEVIATION:TRANSMITTER:AVERAGE:THRUPUT:DLAY?` might return "0,125" <0> indicates measurement is disabled (turned off) for limit comparison, and the upper limit for Transmitter Average throughput delay is <125>.

[SENSe]:P25:SUMMary:LIMIt:MF

Sets or queries for the On/Off state of limit comparison and the upper limit value for Modulation Fidelity. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions Measurement view: P25 Summary

Group Sense commands

Syntax

```
[SENSe]:P25:SUMMary:LIMIt:MF <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:MF?
```

Arguments <NR1>=0 to deselect the measurement and 1 to select the measurement.

<NR2>=0 to 4

Returns <NR1>,<NR2>

Examples `[SENSE]:P25:SUMMARY:LIMIT:MF?` might return "1,4" <1> indicates measurement is enabled (turned on) for limit comparison, and the upper limit for Modulation Fidelity is <4>.

[SENSe]:P25:SUMMary:LIMIt:RF:OUTPower

Sets or queries for the On/Off state of limit comparison and the upper limit value for RF Output Power. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions Measurement view: P25 Summary

Group Sense commands

Syntax
```
[SENSe]:P25:SUMMary:LIMIt:RF:OUTPower <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:RF:OUTPower?
```

Arguments <NR1>=0 to deselect the measurement and 1 to select the measurement.

<NR2>=0 to 3

Returns <NR1>,<NR2>

Examples `[SENSE]:P25:SUMMARY:LIMIT:RF:OUTPOWER?` might return "1,3" <1> indicates measurement is enabled (turned on) for limit comparison, and the upper limit for RF Output Power is <3>.

[SENSe]:P25:SUMMary:LIMIt:SRA

Sets or queries for the On/Off state of limit comparison and the upper limit value for Symbol Rate Accuracy. The limit comparison can be enabled (turned on) or disabled (turned off).

Conditions Measurement view: P25 Summary

Group Sense commands

Syntax
```
[SENSe]:P25:SUMMary:LIMIt:SRA <NR1>,<NR2>
[SENSe]:P25:SUMMary:LIMIt:SRA?
```

Arguments <NR1>=0 to deselect the measurement and 1 to select the measurement.

<NR2>=0 to 10

Returns <NR1>,<NR2>

Examples `[SENSE]:P25:SUMMARY:LIMIT:SRA?` might return "1,10" <1> indicates measurement is enabled (turned on) for limit comparison, and the upper limit for Symbol Rate Accuracy is <10>.

[SENSe]:P25:SYMBols:POINts

Sets or queries the Points/Symbols to display in the P25 measurement view.

Conditions Measurement view: Any P25 measurement display

Group Sense commands

Syntax

```
[SENSe]:P25:SYMBols:POINts { ONE | TWO | FOUR | EIGHt }
[SENSe]:P25:SYMBols:POINts?
```

Arguments ONE set the Points/Symbol trace setting to 1.

TWO set the Points/Symbol trace setting to 2.

FOUR set the Points/Symbol trace setting to 4.

EIGHt set the Points/Symbol trace setting to 8.

Returns ONE means the Points/Symbol trace setting is set to 1.

TWO means the Points/Symbol trace setting is set to 2.

FOUR means the Points/Symbol trace setting to is set 4.

EIGH means the Points/Symbol trace setting to is set 8.

Examples `[SENSE]:P25:SYMBOLS:POINTS?` might return 4, indicating that the Points/Symbol value is set to 4.

SENSe:P25:TIMe:UNITs

This command sets or queries the time units as Seconds or Symbols in the P25 Power vs Time display.

Conditions Measurement view: P25 Power vs Time

Group Sense commands

Syntax

```
SENSe:P25:TIMe:UNITs { SYMB | SEC }
SENSe:P25:TIMe:UNITs?
```

Arguments SYMBol sets the time units to Symbols.

SEConds sets the time units to Seconds.

Returns SYMB means the time units are set to Symbols.

SEC means the time units are set to Seconds.

Examples `SENSE:P25:TIME:UNITS?` might return SEC, indicating that the time units are set to Seconds.

[SENSe]:P25:TIMe:UNITs

Sets or queries the fundamental unit of time for the P25 measurements.

Conditions Measurement view: Any P25 measurement display

Group Sense commands

Syntax
```
[SENSe]:P25:TIMe:UNITs { SEC | SYMB }
[SENSe]:P25:TIMe:UNITs?
```

Arguments SEConds sets the unit of time to seconds.

SYMBols sets the unit of time to symbols.

Returns SEC means that the unit if time is set to seconds.

SYMB means that the unit if time is set to symbols.

Examples `[SENSE]:P25:TIME:UNITS?` might return SEC, indicating that the unit of time is set to seconds.

[SENSe]:PHVTime:CLEar:RESults (No Query Form)

Restarts multi-trace functions (Average and Max/Min Hold).

Conditions Measurement views: Phase versus Time

Group Sense commands

Syntax `[SENSe]:PHVTime:CLEar:RESults`

Arguments None

Examples `SENSE:PHVTIME:CLEAR:RESULTS` restarts multi-trace functions.

[SENSe]:PHVTime:FREQuency:CENTer

Sets or queries the center frequency in the Phase versus Time measurement.

NOTE. *The center, start and stop frequencies are set interlocking each other with the following relationships: (start frequency) = (center frequency) - (span)/2 and (stop frequency) = (center frequency) + (span)/2.*

Conditions Measurement views: Phase versus Time

Group Sense commands

Syntax
```
[SENSe]:PHVTime:FREQuency:CENTer <value>
[SENSe]:PHVTime:FREQuency:CENTer?
```

Related Commands [SENSe]:PHVTime:FREQuency:STARt, [SENSe]:PHVTime:FREQuency:STOP

Arguments `<value> ::  <NRf>` specifies the center frequency.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `SENSE:PHVTIME:FREQUENCY:CENTER 7.5GHz` sets the center frequency to 7.5 GHz.

[SENSe]:PHVTime:FREQuency:SPAN

Sets or queries the frequency span in the Phase versus Time measurement.

Conditions Measurement views: Phase versus Time

Group Sense commands

Syntax
```
[SENSe]:PHVTime:FREQuency:SPAN <value>
[SENSe]:PHVTime:FREQuency:SPAN?
```

Arguments <value> :: <NRf> is the frequency span.
Range: 10 Hz to 40 MHz (Standard) / 110 MHz (Option 110)

Examples SENSE:PHVTIME:FREQUENCY:SPAN 20MHz sets the span to 20 MHz.

[SENSe]:PHVTime:FREQuency:STARt

Sets or queries the measurement start frequency (left edge on the graph) in the Phase versus Time measurement.

The center, start and stop frequencies are set interlocking each other. Refer to the [SENSe]:PHVTime:FREQuency:CENTer command.

Conditions Measurement views: Phase versus Time

Group Sense commands

Syntax
```
[SENSe]:PHVTime:FREQuency:STARt <value>
[SENSe]:PHVTime:FREQuency:STARt?
```

Related Commands [SENSe]:PHVTime:FREQuency:STOP

Arguments <value> :: <NRf> is the measurement start frequency.
Range: (center frequency) ± (span)/2.

Examples SENSE:PHVTIME:FREQUENCY:START 6.95GHz sets the start frequency to 6.95 GHz.

[SENSe]:PHVTime:FREQuency:STEP

Sets or queries the frequency step size (the amount per press by which the up or down key changes the setting value). Programming a specified step size sets [SENSe]:PHVTime:FREQuency:STEP:AUTO OFF.

Conditions Measurement views: Phase versus Time

Group Sense commands

Syntax

```
[SENSe]:PHVTime:FREQuency:STEP <value>
[SENSe]:PHVTime:FREQuency:STEP?
```

Related Commands [SENSe]:PHVTime:FREQuency:STEP:AUTO

Arguments `<value> :: <NRf>` specifies the frequency step size.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `SENSE:PHVTIME:FREQUENCY:STEP 1.5kHz` sets the step size to 1.5 kHz.

[SENSe]:PHVTime:FREQuency:STEP:AUTO

Determines whether to set the frequency step size automatically or manually.

Conditions Measurement views: Phase versus Time

Group Sense commands

Syntax

```
[SENSe]:PHVTime:FREQuency:STEP:AUTO { OFF | ON | 0 | 1 }
[SENSe]:PHVTime:FREQuency:STEP:AUTO?
```

Arguments `OFF` or `0` specifies that the frequency step size is set manually using the [SENSe]:PHVTime:FREQuency:STEP command.

ON or 1 specifies that the frequency step size is set automatically.

Examples SENSE:PHVTIME:BANDWIDTH:AUTO ON sets the frequency step size automatically.

[SENSe]:PHVTime:FREQuency:STOP

Sets or queries the measurement stop frequency (right edge of the graph) in the Phase versus Time measurement.

The center, start and stop frequencies are set interlocking each other. Refer to the [SENSe]:PHVTime:FREQuency:CENTer command.

Conditions Measurement views: Phase versus Time

Group Sense commands

Syntax
```
[SENSe]:PHVTime:FREQuency:STOP <value>
[SENSe]:PHVTime:FREQuency:STOP?
```

Related Commands [SENSe]:PHVTime:FREQuency:STARt

Arguments <value> :: <NRf> is the measurement stop frequency.
Range: (center frequency) ± (span)/2.

Examples SENSE:PHVTIME:FREQUENCY:STOP 7.05GHz sets the stop frequency to 7.05 GHz.

[SENSe]:PHVTime:MAXTracepoints

Sets or queries the maximum trace points in the Phase versus Time measurement.

Conditions Measurement views: Phase versus Time

Group Sense commands

Syntax
```
[SENSe]:PHVTime:MAXTracepoints { ONEK | TENK | HUNDredk |
NEVerdecimate }
[SENSe]:PHVTime:MAXTracepoints?
```

Arguments

`ONEK` sets the maximum trace points to 1 k.

`TENK` sets the maximum trace points to 10 k.

`HUNDredk` sets the maximum trace points to 100 k.

`NEVerdecimate` never decimates the trace points.

Examples

`SENSE:PHVTIME:MAXTRACEPOINTS TENK` sets the maximum trace points to 10 k.

[SENSe]:PM:PHASe:OFFSet

Sets or queries the phase offset in the PM measurement.

Conditions Measurement views: PM

Group Sense commands

Syntax

```
[SENSe]:PM:PHASe:OFFSet <value>
[SENSe]:PM:PHASe:OFFSet?
```

Arguments

`<value> :: <NRf>` specifies the phase offset. Range: –180 to +180 °.

Examples

`SENSE:PM:PHASE:OFFSET 10deg` sets the phase offset to 10 °.

[SENSe]:PM:PHASe:OFFSet:MARKer (No Query Form)

Sets the phase offset from the selected marker location in the PM measurement.

Conditions Measurement views: PM

Group Sense commands

Syntax

```
[SENSe]:PM:PHASe:OFFSet:MARKer
```

Arguments None.

Examples `SENSE:PM:PHASe:OFFSET:MARKER` sets the phase offset from the selected Marker position.

[SENSe]:PM:PHASe:SEARch:AUTO

Determines whether to detect the phase automatically or manually in the PM measurement.

Conditions Measurement views: PM

Group Sense commands

Syntax
```
[SENSe]:PM:PHASe:SEARch:AUTO { OFF | ON | 0 | 1 }
[SENSe]:PM:PHASe:SEARch:AUTO?
```

Arguments `OFF` or `0` specifies that the phase offset is set manually using the command[SENSe]:PM:PHASe:OFFSet.

`ON` or `1` specifies that the phase is detected automatically.

Examples `SENSE:PM:PHASE:SEARCH:AUTO ON` specifies that the phase is detected automatically.

[SENSe]:PNOise:AVERage:COUNt

Sets or queries the number of traces to combine for averaging in the phase noise measurement. This command is effective when [SENSe]:PNOise:AVERage:ENABleis set to ON.

Conditions Measurement views: Phase noise

Group Sense commands

Syntax
```
[SENSe]:PNOise:AVERage:COUNt <number>
[SENSe]:PNOise:AVERage:COUNt?
```

Arguments `<number> ::` `<NR1>` specifies the average count. Range: 2 to 10000.

Examples `SENSE:PNOISE:AVERAGE:COUNT 64` sets the average count to 64.

[SENSe]:PNOise:AVERage:ENABle

Determines whether to enable averaging trace in the phase noise measurement.

Conditions Measurement views: Phase noise

Group Sense commands

Syntax

```
[SENSe]:PNOise:AVERage:ENABle { OFF | ON | 0 | 1 }
[SENSe]:PNOise:AVERage:ENABle?
```

Arguments `OFF` disables averaging trace.

`ON` enables averaging trace.

Examples `SENSE:PNOISE:AVERAGE:ENABLE ON` enables averaging trace.

[SENSe]:PNOise:CARRier:FREQuency:TRACk

Determines whether to enable or disable tracking the carrier frequency in the phase noise measurement.

Conditions Measurement views: Phase noise

Group Sense commands

Syntax

```
[SENSe]:PNOise:CARRier:FREQuency:TRACk { OFF | ON | 0 | 1 }
[SENSe]:PNOise:CARRier:FREQuency:TRACk?
```

Arguments `OFF` or `0` disables tracking the carrier frequency.

`ON` or `1` enables tracking the carrier frequency.

Examples `SENSE:PNOISE:CARRIER:FREQUENCY:TRACK ON` enables tracking the carrier frequency.

[SENSe]:PNOise:CARRier:THReshold

Sets or queries the threshold level to detect the carrier in the phase noise measurement.

Conditions Measurement views: Phase noise

Group Sense commands

Syntax
```
[SENSe]:PNOise:CARRier:THReshold <value>
[SENSe]:PNOise:CARRier:THReshold?
```

Arguments <value> :: <NRf> specifies the threshold level above which the input signal is determined to be a carrier. Range: -60 to 0 dBm.

Examples SENSE:PNOISE:CARRIER:THRESHOLD -25 sets the carrier threshold level to -25 dB.

[SENSe]:PNOise:CLEar:RESults (No Query Form)

Restarts the average process, clearing average data and counter.

Conditions Measurement views: Phase noise

Group Sense commands

Syntax
```
[SENSe]:PNOise:CLEar:RESults
```

Arguments None

Examples SENSE:PNOISE:CLEAR:RESULTS restarts the average process.

[SENSe]:PNOise:FREQuency:INTegration:OFFSet:STARt

Sets or queries the start offset frequency for integration in the phase noise measurement.

Conditions Measurement views: Phase noise

Group Sense commands

Syntax
```
[SENSe]:PNOise:FREQuency:INTegration:OFFSet:STARt <value>
[SENSe]:PNOise:FREQuency:INTegration:OFFSet:STARt?
```

Arguments <value> :: <NRf> specifies the start offset frequency for integration. It must be less than the stop offset frequency.
Range: 10 Hz to Stop Offset Frequency - 1 Hz.

Examples SENSE:PNOISE:FREQUENCY:INTEGRATION:OFFSET:START 100kHz sets the start offset frequency for integration to 100 kHz.

[SENSe]:PNOise:FREQuency:INTegration:OFFSet:STOP

Sets or queries the stop offset frequency for integration in the phase noise measurement.

Conditions Measurement views: Phase noise

Group Sense commands

Syntax
```
[SENSe]:PNOise:FREQuency:INTegration:OFFSet:STOP <value>
[SENSe]:PNOise:FREQuency:INTegration:OFFSet:STOP?
```

Arguments <value> :: <NRf> specifies the stop offset frequency for integration. It must be greater than the start offset frequency.
Range: 11 Hz to 1 GHz.

Examples SENSE:PNOISE:FREQUENCY:INTEGRATION:OFFSET:STOP 100MHz sets the stop offset frequency for integration to 100 MHz.

[SENSe]:PNOise:FREQuency:PLOT:OFFSet:STARt

Sets or queries the start offset frequency for plotting the phase noise trace.

Conditions Measurement views: Phase noise

Group Sense commands

Syntax
```
[SENSe]:PNOise:FREQuency:PLOT:OFFSet:STARt <value>
[SENSe]:PNOise:FREQuency:PLOT:OFFSet:STARt?
```

Arguments <value> :: <NRf> specifies the start offset frequency for the trace plot. It must be less than the stop offset frequency.
Range: 10 Hz to 100 MHz in a tenfold sequence.

Examples SENSE:PNOISE:FREQUENCY:PLOT:OFFSET:START 100kHz sets the start offset frequency for plot to 100 kHz.

[SENSe]:PNOise:FREQuency:PLOT:OFFSet:STOP

Sets or queries the stop offset frequency for plotting the phase noise trace.

Conditions Measurement views: Phase noise

Group Sense commands

Syntax
```
[SENSe]:PNOise:FREQuency:PLOT:OFFSet:STOP <value>
[SENSe]:PNOise:FREQuency:PLOT:OFFSet:STOP?
```

Arguments <value> :: <NRf> specifies the stop offset frequency for plot. It must be greater than the start offset frequency.
Range: 100 Hz to 1 GHz in a tenfold sequence.

Examples SENSE:PNOISE:FREQUENCY:PLOT:OFFSET:STOP 100MHz sets the stop offset frequency for plot to 100 MHz.

[SENSe]:PNOise:OPTimization

Sets or queries the method of optimizing the gain and input bandwidth in the phase noise measurement.

Conditions Measurement views: Phase noise

Group Sense commands

Syntax

```
[SENSe]:PNOise:OPTimization { DRANge | SPEed }
[SENSe]:PNOise:OPTimization?
```

Arguments

`DRANge` optimizes the gain and input bandwidth to maximize the dynamic range.

`SPEed` optimizes the gain and input bandwidth to speed the measurement.

Examples

`SENSE:PNOISE:OPTIMIZATION DRANge` optimizes the gain and input bandwidth to maximize the dynamic range.

[SENSe]:POWer:UNITs

Sets or queries the fundamental unit of power.

Conditions

Measurement views: All

Group

Sense commands

Syntax

```
[SENSe]:POWer:UNITs { DBM | DBV | VOLTs | WATTs | DBUW | DBW
| DBUV | DBMV | DBUA | DBUV_M | DBUA_M | AMPS }
[SENSe]:POWer:UNITs?
```

Arguments

The following table lists the arguments.

Table 2-35: Power units

Argument	Power unit
DBM	dBm
DBV	dBV
VOLTs	Volts
WATTs	Watts
DBUW	dBµW
DBW	dBW
DBUV	dBµV
DBMV	dBmV
DBUA	dBµA
DBUV_M	dBµV/m
DBUA_M	dBµA/m
AMPS	Amps

NOTE. Select dBµV/m or dBµA/m unit when using an antenna table.

Examples `SENSE:POWER:UNITS DBM` specifies the fundamental unit of power as dBm.

[SENSe]:PULSe:ANALyze:FDOMain:MEASurement:TIME:ALENgth

Sets or queries the absolute amount of time included in the Frequency-Domain Linearity type measurements.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax

```
[SENSe]:PULSe:ANALyze:FDOMain:MEASurement:TIME:ALENgth
<value>
[SENSe]:PULSe:ANALyze:FDOMain:MEASurement:TIME:ALENgth?
```

Arguments `<value> ::= <NRf>` specifies the absolute time in seconds for the measurement time.

Range: 1 ns to 100 seconds

Examples `[SENSE]:PULSE:ANALYZE:FDOMAIN:MEASUREMENT:TIME:ALENGTH 1E-6` sets the measurement time to 1 µs.

[SENSe]:PULSe:ANALyze:FDOMain:MEASurement:TIME:METHod

Sets or queries the measurement time method for the Frequency-Domain Linearity type measurements.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax

```
[SENSe]:PULSe:ANALyze:FDOMain:MEASurement:TIME:METHod {
ABSolute | RELative }
[SENSe]:PULSe:ANALyze:FDOMain:MEASurement:TIME:METHod?
```

Arguments `ABSolute` specifies the absolute time in seconds based on rising edge of the pulse.

`RELative` specifies the relative time in percent based on the pulse width and the center of the pulse.

Examples `[SENSE]:PULSE:ANALYZE:FDOMAIN:MEASUREMENT:TIME:METHOD REL` sets the time method to Relative.

[SENSe]:PULSe:ANALyze:FDOMain:MEASurement:TIME:RLENgth

Sets or queries the relative amount of time included in the Frequency-Domain Linearity type measurements.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax
```
[SENSe]:PULSe:ANALyze:FDOMain:MEASurement:TIME:RLENgth
<value>
[SENSe]:PULSe:ANALyze:FDOMain:MEASurement:TIME:RLENgth?
```

Arguments `<value> :: <NRf>` specifies the relative time in percent for the measurement time based on the pulse width and centered of the pulse.

Range: 10% to 100%

Examples `[SENSE]:PULSE:ANALYZE:FDOMAIN:MEASUREMENT:TIME:RLENGTH 70` sets the measurement time to 70% of the pulse width.

[SENSe]:PULSe:ANALyze:FDOMain:MEASurement:TIME:RLEVel

Sets or queries the measurement reference level for Frequency-Domain Linearity measurements.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax
```
[SENSe]:PULSe:ANALyze:FDOMain:MEASurement:TIME:RLEVel
<value>
```

```
[SENSe]:PULSe:ANALyze:FDOMain:MEASurement:TIME:RLEVel?
```

Arguments <value> :: <NRf> specifies the reference level in percentage as the location on the voltage straight-line approximation of the pulse rising edge.

Range: 50% to 100%

Examples `[SENSE]:PULSE:ANALYZE:FDOMAIN:MEASUREMENT:TIME:RLEVEL 60` sets the reference level to 60% of the rising edge.

[SENSe]:PULSe:ANALyze:FDOMain:MEASurement:TIME:STARt

Sets or queries the start time for the Frequency-Domain Linearity type measurements. The start time is the amount of time wait after the rising-edge reference before starting to measure.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax

```
[SENSe]:PULSe:ANALyze:FDOMain:MEASurement:TIME:STARt <value>
[SENSe]:PULSe:ANALyze:FDOMain:MEASurement:TIME:STARt?
```

Arguments <value> :: <NRf> specifies the start time in seconds from the reference level of the rising edge of the pulse.

Range: –100 ms to 100 ms

Examples `[SENSE]:PULSE:ANALYZE:FDOMAIN:MEASUREMENT:TIME:START 1E-6` sets the start time to 1 µs.

[SENSe]:PULSe:ANALyze:IRESponse:CORRection:AMPLitude[:STATe]

Enables or disables the corrections for window and for delay of the highest side lobe in the Pulse Impulse Response measurements.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax

```
[SENSe]:PULSe:ANALyze:IRESponse:CORRection:AMPLitude[:STATe]
{ OFF | ON | 0 | 1 }
[SENSe]:PULSe:ANALyze:IRESponse:CORRection:AMPLitude[:
STATe]?
```

Arguments

`OFF` or `0` disables amplitude correction in Impulse Response measurements.

`ON` or `1` enables amplitude correction in Impulse Response measurements.

Examples

`[SENSE]:PULSE:ANALYZE:IRESPONSE:CORRECTION:AMPLITUDE[:STATE] 1` enables amplitude correction.

[SENSe]:PULSe:ANALyze:IRESponse:KOTime

Sets or queries the Keep-out time in the Pulse Impulse Response measurements. This specifies the width of time around the main lobe that lobes within this time zone are not eligible to be the "highest side lobe."

Conditions

Measurement views: Pulse statistics, Pulse table, Pulse trace

Group

Sense commands

Syntax

```
[SENSe]:PULSe:ANALyze:IRESponse:KOTime <value>
[SENSe]:PULSe:ANALyze:IRESponse:KOTime?
```

Arguments

`<value> ::  <NRf>` specifies the Keep-out time in seconds.

Range: 0 to 100 seconds.

Examples

`[SENSE]:PULSE:ANALYZE:IRESPONSE:KOTIME 0.001` sets the keep-out time to 1 ms.

[SENSe]:PULSe:ANALyze:IRESponse:MEASurement:TIME:ALENgth

Sets or queries the absolute amount of time included in the Impulse Response type measurements.

Conditions

Measurement views: Pulse statistics, Pulse table, Pulse trace

Group

Sense commands

Syntax

```
[SENSe]:PULSe:ANALyze:IRESponse:MEASurement:TIME:ALENgth
<value>
[SENSe]:PULSe:ANALyze:IRESponse:MEASurement:TIME:ALENgth?
```

Arguments

<value> :: <NRf> specifies the absolute time in seconds for the measurement time.

Range: 1 ns to 100 seconds.

Examples

[SENSE]:PULSE:ANALYZE:IRESPONSE:MEASUREMENT:TIME:ALENGTH 1E-6 sets the measurement time to 1 μs.

[SENSe]:PULSe:ANALyze:IRESponse:MEASurement:TIME:METHod

Sets or queries the measurement time method for the Impulse Response type measurements.

Conditions

Measurement views: Pulse statistics, Pulse table, Pulse trace

Group

Sense commands

Syntax

```
[SENSe]:PULSe:ANALyze:IRESponse:MEASurement:TIME:METHod {
ABSolute | RELative }
[SENSe]:PULSe:ANALyze:IRESponse:MEASurement:TIME:METHod?
```

Arguments

ABSolute specifies the absolute time in seconds based on rising edge of the pulse.

RELative specifies the relative time in percent based on the pulse width and the center of the pulse.

Examples

[SENSE]:PULSE:ANALYZE:IRESPONSE:MEASUREMENT:TIME:METHOD REL sets the time method to Relative.

[SENSe]:PULSe:ANALyze:IRESponse:MEASurement:TIME:RLENgth

Sets or queries the relative amount of time included in the Impulse Response type measurements.

Conditions

Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax
```
[SENSe]:PULSe:ANALyze:IRESponse:MEASurement:TIME:RLENgth
<value>
[SENSe]:PULSe:ANALyze:IRESponse:MEASurement:TIME:RLENgth?
```

Arguments <value> :: <NRf> specifies the relative time in percent for the measurement time based on the pulse width and centered of the pulse.

Range: 10% to 100%.

Examples `[SENSE]:PULSE:ANALYZE:IRESPONSE:MEASUREMENT:TIME:RLENGTH 70` sets the measurement time to 70% of the pulse width.

[SENSe]:PULSe:ANALyze:IRESponse:MEASurement:TIME:RLEVel

Sets or queries the measurement reference level for Impulse Response measurements.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax
```
[SENSe]:PULSe:ANALyze:IRESponse:MEASurement:TIME:RLEVel
<value>
[SENSe]:PULSe:ANALyze:IRESponse:MEASurement:TIME:RLEVel?
```

Arguments <value> :: <NRf> specifies the reference level in percentage as the location on the voltage straight-line approximation of the pulse rising edge.

Range: 50% to 100%.

Examples `[SENSE]:PULSE:ANALYZE:IRESPONSE:MEASUREMENT:TIME:RLEVEL 60` sets the reference level to 60% of the rising edge.

[SENSe]:PULSe:ANALyze:IRESponse:MEASurement:TIME:STARt

Sets or queries the start time for the Impulse Response type measurements. The start time is the amount of time wait after the rising-edge reference before starting to measure.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax
```
[SENSe]:PULSe:ANALyze:IRESponse:MEASurement:TIME:STARt
<value>
[SENSe]:PULSe:ANALyze:IRESponse:MEASurement:TIME:STARt?
```

Arguments `<value> ::= <NRf>` specifies the start time in seconds from the reference level of the rising edge of the pulse.

Range: –100 ms to 100 ms.

Examples `[SENSE]:PULSE:ANALYZE:IRESPONSE:MEASUREMENT:TIME:START 1E-6` sets the start time to 1 µs.

[SENSe]:PULSe:ANALyze:LEVel

Sets or queries how to determine the 50% level for the pulsed RF measurements.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax
```
[SENSe]:PULSe:ANALyze:LEVel { VOLTage | POWer }
[SENSe]:PULSe:ANALyze:LEVel?
```

Arguments `VOLTage` uses -6 dB to determine the 50% level.

`POWer` uses -3 dB to determine the 50% level.

Examples `SENSe:PULSe:ANALyze:LEVel POWer` sets RF measurements to use -3 dB to determine the 50% level.

[SENSe]:PULSe:ANALyze:LEVel:FIFTy

Sets or queries how to determine the 50% level for the pulsed RF measurements. This command is equivalent to the[SENSe]:PULSe:ANALyze:LEVel command.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax
```
[SENSe]:PULSe:ANALyze:LEVel:FIFTy { VOLTage | POWer }
[SENSe]:PULSe:ANALyze:LEVel:FIFTy?
```

Arguments `VOLTage` uses -6 dB to determine the 50% level.

`POWer` uses -3 dB to determine the 50% level.

Examples `SENSE:PULSE:ANALYZE:LEVEL:FIFTY POWer` sets RF measurements to use -3 dB to determine the 50% level.

[SENSe]:PULSe:ANALyze:LEVel:HUNDred

Sets or queries how to determine the 100% level in the pulsed RF measurements. This command is equivalent to the [SENSe]:PULSe:ANALyze:POINt:LOCationcommand.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax
```
[SENSe]:PULSe:ANALyze:LEVel:HUNDred { AVERage | INDependent }
[SENSe]:PULSe:ANALyze:LEVel:HUNDred?
```

Related Commands

Arguments `AVERage` uses the average amplitude calculated for the pulse-on as the 100% reference to measure the rise and fall times.

`INDependent` uses the amplitudes at the beginning and end of the pulse-on as the 100% references to measure the rise and fall times, respectively

Examples `SENSE:PULSE:ANALYZE:LEVEL:HUNDRED AVERage` uses pulse average amplitude to set the 100% level.

[SENSe]:PULSe:ANALyze:OVERshoot:MEASurement:TIME:ALENgth

Sets or queries the absolute amount of time included in the Overshoot measurements.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax

```
[SENSe]:PULSe:ANALyze:OVERshoot:MEASurement:TIME:ALENgth
<value>
[SENSe]:PULSe:ANALyze:OVERshoot:MEASurement:TIME:ALENgth?
```

Arguments `<value> ::  <NRf>` specifies the absolute time in seconds for the measurement time.

Range: 1 ns to 100 seconds

Examples `[SENSE]:PULSE:ANALYZE:OVERSHOOT:MEASUREMENT:TIME:ALENGTH 1E-6` sets the measurement time to 1 μs.

[SENSe]:PULSe:ANALyze:OVERshoot:MEASurement:TIME:METHod

Sets or queries the measurement time method for the Overshoot measurements.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax

```
[SENSe]:PULSe:ANALyze:OVERshoot:MEASurement:TIME:METHod {
ABSolute | RELative }
[SENSe]:PULSe:ANALyze:OVERshoot:MEASurement:TIME:METHod?
```

Arguments `ABSolute` specifies the absolute time in seconds based on rising edge of the pulse.

`RELative` specifies the relative time in percent based on the pulse width and the center of the pulse.

Examples `[SENSE]:PULSE:ANALYZE:OVERSHOOT:MEASUREMENT:TIME:METHOD REL` sets Overshoot measurements to use the relative time method.

[SENSe]:PULSe:ANALyze:OVERshoot:MEASurement:TIME:RLENgth

Sets or queries the relative amount of time included in the Overshoot measurements.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax
```
[SENSe]:PULSe:ANALyze:OVERshoot:MEASurement:TIME:RLENgth
<value>
[SENSe]:PULSe:ANALyze:OVERshoot:MEASurement:TIME:RLENgth?
```

Arguments <value> :: <NRf> specifies the relative time in percent for the measurement time based on the pulse width and centered of the pulse.

Range: 0.01% to 100%.

Examples `[SENSE]:PULSE:ANALYZE:OVERSHOOT:MEASUREMENT:TIME:RLENGTH 70` sets the measurement time to 70% of the pulse width.

[SENSe]:PULSe:ANALyze:OVERshoot:MEASurement:TIME:RLEVel

Sets or queries the measurement reference level for the Overshoot measurements.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax
```
[SENSe]:PULSe:ANALyze:OVERshoot:MEASurement:TIME:RLEVel
<value>
[SENSe]:PULSe:ANALyze:OVERshoot:MEASurement:TIME:RLEVel?
```

Arguments <value> :: <NRf> specifies the reference level in percentage as the location on the voltage straight-line approximation of the pulse rising edge.

Range: 50% to 100%.

Examples `[SENSE]:PULSE:ANALYZE:OVERSHOOT:MEASUREMENT:TIME:RLEVEL 60` sets the reference level to 60% of the rising edge.

[SENSe]:PULSe:ANALyze:OVERshoot:MEASurement:TIME:STARt

Sets or queries the start time for the Overshoot measurements. The start time is the amount of time wait after the rising-edge reference before starting to measure.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax
```
[SENSe]:PULSe:ANALyze:OVERshoot:MEASurement:TIME:STARt
<value>
[SENSe]:PULSe:ANALyze:OVERshoot:MEASurement:TIME:STARt?
```

Arguments <value> :: <NRf> specifies the start time in seconds from the reference level of the rising edge of the pulse.

Range: –100 ms to 100 ms.

Examples `[SENSE]:PULSE:ANALYZE:OVERSHOOT:MEASUREMENT:TIME:START 1E-6` sets the start time to 1 µs.

[SENSe]:PULSe:ANALyze:PMLocation

Sets or queries the phase measurement location (the position along the pulse tops where the phase is measured) in the pulse-pulse phase measurement.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax
```
[SENSe]:PULSe:ANALyze:PMLocation <value>
[SENSe]:PULSe:ANALyze:PMLocation?
```

Arguments <value> :: <NRf> specifies the pulse-pulse phase measurement location.
Range: 5 ns to 100 ms.

Examples `SENSE:PULSE:ANALYZE:PMLOCATION 1.5ms` sets the phase measurement location to 1.5 ms.

[SENSe]:PULSe:ANALyze:POINt:LOCation

Sets or queries the point location method in the pulsed RF measurements.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax

```
[SENSe]:PULSe:ANALyze:POINt:LOCation { AVERage | INDependent
}
[SENSe]:PULSe:ANALyze:POINt:LOCation?
```

Arguments `AVERage` uses the average amplitude calculated for the pulse-on as the 100% reference to measure the rise and fall times.

`INDependent` uses the amplitudes at the beginning and end of the pulse-on as the 100% references to measure the rise and fall times, respectively.

Examples `SENSE:PULSE:ANALYZE:POINT:LOCATION AVERage` uses pulse average amplitude to locate points.

[SENSe]:PULSe:ANALyze:RFALl

Sets or queries the threshold levels to measure the rise/fall time.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax

```
[SENSe]:PULSe:ANALyze:RFALl { WIDE | NARRow }
[SENSe]:PULSe:ANALyze:RFALl?
```

Arguments `WIDE` selects 10 - 90% to measure the rise/fall time.

`NARRow` selects 20 - 80% to measure the rise/fall time.

Examples `SENSE:PULSE:ANALYZE:RFALL WIDE` selects 10 - 90% to measure the rise/fall time.

[SENSe]:PULSe:ANALyze:RIPPle

Sets or queries the ripple portion of the pulse top (how much of the beginning and end of the pulse top to exclude from the ripple calculation).

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax
```
[SENSe]:PULSe:ANALyze:RIPPle <value>
[SENSe]:PULSe:ANALyze:RIPPle?
```

Arguments <value> :: <NRf> specifies the ripple portion of the pulse top.
Range: 10 to 100% in 1% steps.

Examples SENSE:PULSE:ANALYZE:RIPPLE 30 sets the ripple portion to 30% of the pulse top.

[SENSe]:PULSe:CARRier:FREQuency? (Query Only)

Queries the carrier frequency.

Conditions Measurement views: Pulse table

Group Sense commands

Syntax
```
[SENSe]:PULSe:CARRier:FREQuency?
```

Related Commands [SENSe]:PULSe:CARRier:OFFSet

Arguments None

Returns <NRf> indicates the actual carrier frequency in Hz.

Examples SENSE:PULSE:CARRIER:FREQUENCY? might return 2.7000001690E+9, indicating that the pulse carrier frequency is 2.700000169 GHz.

[SENSe]:PULSe:CARRier:OFFSet

Sets or queries the carrier frequency offset.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax
```
[SENSe]:PULSe:CARRier:OFFSet <value>
[SENSe]:PULSe:CARRier:OFFSet?
```

Related Commands [SENSe]:PULSe:CARRier:SEARch

Arguments <value> :: <NRf> specifies the carrier frequency offset.
Range: -50 kHz to +50 kHz.

Examples SENSE:PULSE:CARRIER:OFFSET 2.5kHz sets the offset frequency to 2.5 kHz to the carrier.

[SENSe]:PULSe:CARRier:SEARch

Sets or queries how to detect the carrier in the pulsed RF measurements.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax
```
[SENSe]:PULSe:CARRier:SEARch { AUTO | MANual }
[SENSe]:PULSe:CARRier:SEARch?
```

Arguments AUTO specifies that the carrier is detected automatically.

MANual specifies that the carrier frequency offset is set manually, using the [SENSe]:PULSe:CARRier:OFFSet command.

Examples SENSE:PULSE:CARRIER:SEARCH AUTO specifies that the carrier is detected automatically.

[SENSe]:PULSe:DETect:MEASurement

Determines whether or not to set the maximum number of pulses to measure within the analysis time.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax

```
[SENSe]:PULSe:DETect:MEASurement { OFF | ON | 0 | 1 }
[SENSe]:PULSe:DETect:MEASurement?
```

Arguments `OFF` or `0` measures all pulses (max. 1000) in the analysis time.

`ON` or `1` specifies that the maximum number of pulses is set manually, using the[SENSe]:PULSe:DETect:NUMBer command.

Examples `SENSE:PULSE:DETECT:MEASUREMENT ON` specifies that the maximum number of pulses is set manually.

[SENSe]:PULSe:DETect:NUMBer

Sets or queries the maximum number of pulses to measure within the analysis time when [SENSe]:PULSe:DETect:MEASurementis On.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax

```
[SENSe]:PULSe:DETect:NUMBer <value>
[SENSe]:PULSe:DETect:NUMBer?
```

Arguments `<value> :: <NRf>` specifies the maximum number of pulses to measure within the analysis time. Range: 1 to 1000.

If the analysis time contains fewer pulses than this number, all of these are measured.

Examples `SENSe:PULSe:DETect:NUMBer 850` sets the maximum number of pulses to 850.

[SENSe]:PULSe:DETect:POWer[:THReshold]

Sets or queries the power threshold to detect pulses.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax

```
[SENSe]:PULSe:DETect:POWer[:THReshold] <value>
[SENSe]:PULSe:DETect:POWer[:THReshold]?
```

Arguments <value> :: <NRf> specifies the power threshold to detect pulses. Range: -70 to 0 dB.

Examples SENSE:PULSE:DETECT:POWER:THRESHOLD -20 sets the power threshold to -20 dB.

[SENSe]:PULSe:DETect:TIMe[:THReshold]

Sets or queries the minimum off-time between pulses.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax

```
[SENSe]:PULSe:DETect:TIMe[:THReshold] <value>
[SENSe]:PULSe:DETect:TIMe[:THReshold]?
```

Arguments <value> :: <NRf> specifies the minimum off-time between pulses. Range: 1 ns to 100 ms.

Examples SENSE:PULSE:DETECT:TIME:THRESHOLD 1.5ms sets the time threshold to 1.5 ms.

[SENSe]:PULSe:FILTer:{BANDwidth|BWIDth}

Sets or queries the filter or acquisition bandwidth when [SENSe]:PULSe:FILTer:MEASurement is set to GAUSsian or NONE (No filter), respectively.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax

```
[SENSe]:PULSe:FILTer:{BANDwidth|BWIDth} <value>
[SENSe]:PULSe:FILTer:{BANDwidth|BWIDth}?
```

Related Commands [SENSe]:PULSe:FILTer:MEASurement

Arguments `<value> :: <NRf>` specifies the filter/acquisition bandwidth depending the [SENSe]:PULSe:FILTer:MEASurement setting. The table below shows the setting range. You can enter any value, but it is rounded up to the next valid number.

[SENSe]:PULSe:FILTer :MEASurement	Range
GAUSsian	100 Hz to 20 MHz (Standard) / 100 Hz to 55 MHz (Option 110) in 1-2-3-5 sequence.
NONE	152, 305, 610, 1.22 k, 2.44 k, 4.88 k, 9.76 k, 19.3 k, 39 k, 78 k, 156 k, 312 k, 625 k, 1.25 M, 2.5 M, 5 M, 10 M, 20 M, 40 MHz, and optionally 60 M and 110 MHz (Option 110).

Examples `SENSE:PULSE:FILTER:BANDWIDTH 10MHz` sets the filter bandwidth to 10 MHz.

[SENSe]:PULSe:FILTer:MEASurement

Sets or queries the measurement filter in the pulsed RF measurements.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax

```
[SENSe]:PULSe:FILTer:MEASurement { GAUSsian | NONE | MAXRtbw
}
```

```
[SENSe]:PULSe:FILTer:MEASurement?
```

Related Commands [SENSe]:PULSe:FILTer:{BANDwidth|BWIDth}

Arguments `GAUSsian` uses the Gaussian filter in the pulsed RF measurements. Use the [SENSe]:PULSe:FILTer:{BANDwidth|BWIDth} command to set the filter bandwidth.

`NONE` uses no filter. Use the [SENSe]:PULSe:FILTer:{BANDwidth|BWIDth} command to set the acquisition bandwidth.

`MAXRtbw` uses no filter. The acquisition bandwidth is fixed to the maximum real-time bandwidth: 40 MHz (Standard) or 110 MHz (Option 110).

Examples `SENSE:PULSE:FILTER:MEASUREMENT GAUSsian` uses the Gaussian filter in the pulsed RF measurements.

[SENSe]:PULSe:FREFerence:AUTO

Determines whether to estimate the pulse frequency reference automatically or manually in the pulsed RF measurements.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax

```
[SENSe]:PULSe:FREFerence:AUTO { OFF | ON | 0 | 1 }
[SENSe]:PULSe:FREFerence:AUTO?
```

Arguments `OFF` or `0` estimates the frequency reference manually. Use the [SENSe]:PULSe:FREFerence:OFFSetcommand to set the frequency offset.

`ON` or `1` estimates the frequency reference automatically.

Examples `SENSE:PULSE:FREFERENCE:AUTO ON` specifies that the frequency reference is estimated automatically.

[SENSe]:PULSe:FREFerence:CSLope

Sets or queries the Chirp Slope.

Chirp Slope is used when the Modulation Type is set to Linear Chirp. When set, this also changes the Chirp Slope to manual mode. When chirp slope is in auto mode, the query returns the estimated chirp slope.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax

```
[SENSe]:PULSe:FREFerence:CSLope <value>
[SENSe]:PULSe:FREFerence:CSLope?
```

Arguments `<value>:=NRf` specifies the chirp slope in Hz/µs.

Range: -100 GHz/µs to 100 GHz/µs.

Examples `[SENSE]:PULSE:FREFERENCE:CSLOPE 1E6` sets the chirp slope to 1 MHz/µs.

[SENSe]:PULSe:FREFerence:CSLope:AUTO

Sets or queries enabling automatic Chirp Slope estimating in the pulsed RF measurements.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax

```
[SENSe]:PULSe:FREFerence:CSLope:AUTO { OFF | ON | 0 | 1 }
[SENSe]:PULSe:FREFerence:CSLope:AUTO?
```

Arguments `OFF` or `0` estimates the chirp slope manually. Use the[SENSe]:PULSe:FREFerence:CSLope command to set the chirp slope.

`ON` or `1` estimates the chirp slope automatically.

Examples `[SENSE]:PULSE:FREFERENCE:CSLOPE:AUTO 1` sets the chirp slope to auto mode.

[SENSe]:PULSe:FREFerence:OFFSet

Sets or queries the frequency reference offset. This command is valid when [SENSe]:PULSe:FREFerence:AUTOis set to OFF.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax

```
[SENSe]:PULSe:FREFerence:OFFSet <value>
[SENSe]:PULSe:FREFerence:OFFSet?
```

Arguments <value> :: <NRf> specifies the frequency reference offset.
Range: -50 kHz to +50 kHz.

Examples SENSE:PULSE:FREFERENCE:OFFSET 2.5kHz sets the frequency offset to 2.5 kHz to the reference.

[SENSe]:PULSe:MODulation:TYPE

Sets or queries the modulation type in the pulsed RF measurements. This command is equivalent to the [SENSe]:REANalyze:NEXTcommand.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax

```
[SENSe]:PULSe:MODulation:TYPE { CWConst | CWCHange | LCHirp
| OTHer }
[SENSe]:PULSe:MODulation:TYPE?
```

Arguments CWConst selects the CW (continuous wave) with constant phase.

CWCHange selects the CW (continuous wave) with changing phase.

LCHirp selects the linear chirp.

OTHer selects other.

Examples SENSE:PULSE:MODULATION:TYPE CWCHange selects the CW with changing phase as the modulation type.

[SENSe]:PULSe:SIGNal:TYPE

Sets or queries the signal type in the pulsed RF measurements.

Conditions Measurement views: Pulse statistics, Pulse table, Pulse trace

Group Sense commands

Syntax
```
[SENSe]:PULSe:SIGNal:TYPE { CWConst | CWCHange | LCHirp |
OTHer}
[SENSe]:PULSe:SIGNal:TYPE?
```

Arguments `CWConst` selects CW (continuous wave) with constant phase.

`CWCHange` selects CW (continuous wave) with changing phase.

`LCHirp` selects linear chirp.

`OTHer` selects other.

Examples `SENSE:PULSE:SIGNAL:TYPE CWCHange` selects the CW with changing phase as the signal type.

[SENSe]:PULSe:STATistics:HISTogram:ORDinate

Sets or queries the Histogram Y unit. This command is valid when DISPlay:PULSe:STATistics:PLOT is set to Histogram.

Conditions Measurement views: Pulse statistics

Group Sense commands

Syntax
```
[SENSe]:PULSe:STATistics:HISTogram:ORDinate { PERCent |
COUNt }
[SENSe]:PULSe:STATistics:HISTogram:ORDinate?
```

Arguments `PERCent` sets the Histogram plot Y axis to the percentage of points.

`COUNt` sets the Histogram plot Y axis to the number of points in the bin.

Examples `[SENSE]:PULSE:STATISTICS:HISTOGRAM:ORDINATE PERCENT` sets the histogram Y axis to percent.

[SENSe]:PULSe:STATistics:HISTogram:POINts

Sets or queries the number of bins in the Histogram plot.

Conditions Measurement views: Pulse statistics

Group Sense commands

Syntax
```
[SENSe]:PULSe:STATistics:HISTogram:POINts <value>
[SENSe]:PULSe:STATistics:HISTogram:POINts?
```

Arguments <value> ::= <NRf> specifies the number of histogram bins.

Range: 5 to 10000

Examples `[SENSE]:PULSE:STATISTICS:HISTOGRAM:POINTS` 10 sets the number of histogram bins to 10.

[SENSe]:PULSe:STATistics:HISTogram:POINts:AUTO

Sets or queries enabling automatic Histogram plot bins count.

Conditions Measurement views: Pulse statistics

Group Sense commands

Syntax
```
[SENSe]:PULSe:STATistics:HISTogram:POINts:AUTO { OFF | ON
| 0 | 1 }
[SENSe]:PULSe:STATistics:HISTogram:POINts:AUTO?
```

Arguments `OFF` or `0` sets the number of bins manually by the [SENSe]:PULSe:STATistics:HISTogram:POINts command.

`ON` or `1` sets the number of bins to the number of pulses/10.

Examples `[SENSE]:PULSE:STATISTICS:HISTOGRAM:POINTS:AUTO 1` sets the number of histogram bins based on number of pulses.

[SENSe]:REANalyze (No Query Form)

Sets all measurements to reanalyze the last acquisition.

NOTE. *SENSe:REANalyze is an overlapped command, which does not finish executing before the next command starts executing. Use the *OPC(?) and *WAI commands to synchronize all pending operations to the execution of this command.*

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze`

Related Commands *OPC, *WAI

Arguments None

Examples `SENSE:REANALYZE` reanalyzes all measurements using the current acquisition.

[SENSe]:REANalyze:ACQuisition:SETTings? (Query Only)

Queries the settings of all acquisitions.

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:ACQuisition:SETTings?`

[SENSe]:REANalyze:ALL (No Query Form)

Replay all items in the current acquisition selection.

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:ALL`

[SENSe]:REANalyze:ALL:LOOP (No Query Form)

Replay all selected records/frames in a continuous loop.

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:ALL:LOOP`

[SENSe]:REANalyze:CURRent (No Query Form)

Replay the currently selected acquisition (or frame).

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:CURRent`

[SENSe]:REANalyze:CURRent:ACQuisition? (Query Only)

Queries the Acquisition record identifier (ID) of the currently selected acquisition (or frame).

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:CURRent:ACQuisition?`

[SENSe]:REANalyze:CURRent:FRAMe? (Query Only)

Queries the Frame identifier (ID) of the currently selected acquisition (or frame).

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:CURRent:FRAMe?`

[SENSe]:REANalyze:CURRent:TIMestamp? (Query Only)

Queries the time stamp of the currently selected acquisition (or frame).

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:CURRent:TIMestamp?`

[SENSe]:REANalyze:DELete:DATA (No Query Form)

Deletes all acquisition data.

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:DELete:DATA`

[SENSe]:REANalyze:FIRSt (No Query Form)

Replays (reanalyzes) the first item in the currently selected acquisition (or frame).

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:FIRSt`

[SENSe]:REANalyze:LAST (No Query Form)

Replays (reanalyzes) the last item in the currently selected acquisition (or frame).

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:LAST`

[SENSe]:REANalyze:NEXT (No Query Form)

Replays (reanalyzes) the next item in the currently selected acquisition (or frame).

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:NEXT`

[SENSe]:REANalyze:PAUSe

Sets or queries the reanalysis paused state.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:REANalyze:PAUSe {OFF | ON | 0 | 1 }
[SENSe]:REANalyze:PAUSe?
```

Arguments `ON` or `1` pauses (blocks) any acquisition until current reanalysis is completed.

`OFF` or `0` disables the reanalysis pause mode.

[SENSe]:REANalyze:PREVious (No Query Form)

Replays (reanalyzes) the previous item in the currently selected acquisition (or frame).

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:REANalyze:PREVious
```

[SENSe]:REANalyze:SELect:ACQuisition:DATA:TYPE

Sets or queries the acquisition data type.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:REANalyze:SELect:ACQuisition:DATA:TYPE {SPECtra | IQ }
[SENSe]:REANalyze:SELect:ACQuisition:DATA:TYPE?
```

Arguments `SPECtra` sets the acquisition data type as DPX spectra data.

`IQ` sets the acquisition date type a IQ data.

Examples `[SENSE]:REANALYZE:SELECT:ACQUISITION:DATA:TYPE IQ` sets the acquisition data type to IQ data.

[SENSe]:REANalyze:SELect:ACQuisition:FIRSt? (Query Only)

Queries the first selectable acquisition Start or Stop value in the currently selected set.

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:SELect:ACQuisition:FIRSt?`

Related Commands [SENSe]:REANalyze:SELect:ACQuisition:FIRSt:TIMestamp?

[SENSe]:REANalyze:SELect:ACQuisition:FIRSt:TIMestamp? (Query Only)

Queries the time stamp of the first selectable acquisition history Start or Stop value in the currently selected acquisition set.

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:SELect:ACQuisition:FIRSt:TIMestamp?`

Related Commands [SENSe]:REANalyze:SELect:ACQuisition:FIRSt?

[SENSe]:REANalyze:SELect:ACQuisition:LAST? (Query Only)

Queries the last selectable acquisition Start or Stop value in the currently selected set.

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:SELect:ACQuisition:LAST?`

Related Commands [SENSe]:REANalyze:SELect:ACQuisition:LAST:TIMestamp?

[SENSe]:REANalyze:SELect:ACQuisition:LAST:TIMestamp? (Query Only)

Queries the time stamp of the last selectable acquisition Start or Stop value in the currently selected set.

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:SELect:ACQuisition:LAST:TIMestamp?`

Related Commands [SENSe]:REANalyze:SELect:ACQuisition:LAST?

[SENSe]:REANalyze:SELect:ACQuisition:STARt

Sets or queries the Acquisition Start value in the currently selected set.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:REANalyze:SELect:ACQuisition:STARt <value>
[SENSe]:REANalyze:SELect:ACQuisition:STARt?
```

Arguments `<value> ::= <NR1>` is an integer in the range from one to the number of acquisitions in the acquisition history.

[SENSe]:REANalyze:SELect:ACQuisition:STARt:TIMestamp? (Query Only)

Queries the time stamp of the current Acquisition Start selection. It is the time stamp of the first frame of the selected acquisition, if FastFrame is enabled.

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:SELect:ACQuisition:STARt:TIMestamp?`

[SENSe]:REANalyze:SELect:ACQuisition:STOP

Sets or queries the Acquisition Stop value in the currently selected set.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:REANalyze:SELect:ACQuisition:STOP <value>
[SENSe]:REANalyze:SELect:ACQuisition:STOP?
```

Arguments <value> :: <NR1> is an integer in the range from one to the number of acquisitions in the acquisition history.

[SENSe]:REANalyze:SELect:ACQuisition:STOP:TIMestamp? (Query Only)

Queries the time stamp of the current Acquisition Stop selection. This is the time stamp of the last frame of the selected acquisition, if FastFrame is enabled.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:REANalyze:SELect:ACQuisition:STOP:TIMestamp?
```

[SENSe]:REANalyze:SELect:ALL (No Query Form)

Selects all acquisitions and frames in the currently selected set.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:REANalyze:SELect:ALL
```

[SENSe]:REANalyze:SELect:FRAMe:FIRSt? (Query Only)

Queries the first available Frame Start or Stop value available in the currently selected set.

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:SELect:FRAMe:FIRSt?`

Related Commands [SENSe]:REANalyze:SELect:FRAMe:FIRSt:TIMestamp?

[SENSe]:REANalyze:SELect:FRAMe:FIRSt:TIMestamp? (Query Only)

Queries the time stamp of the first available frame of the first available Acquisition Start or Stop value available in the currently selected set.

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:SELect:FRAMe:FIRSt:TIMestamp?`

Related Commands [SENSe]:REANalyze:SELect:FRAMe:FIRSt?

[SENSe]:REANalyze:SELect:FRAMe:LAST? (Query Only)

Queries the last available Frame Start or Stop value available in the currently selected set.

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:SELect:FRAMe:LAST?`

Related Commands [SENSe]:REANalyze:SELect:FRAMe:LAST:TIMestamp?

[SENSe]:REANalyze:SELect:FRAMe:LAST:TIMestamp? (Query Only)

Queries the time stamp of the last available frame of the last available Acquisition Start or Stop value available in the currently selected set.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:REANalyze:SELect:FRAMe:LAST:TIMestamp?
```

Related Commands [SENSe]:REANalyze:SELect:FRAMe:LAST?

[SENSe]:REANalyze:SELect:FRAMe:STARt

Sets or queries the Frame Start value in the currently selected set.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:REANalyze:SELect:FRAMe:STARt <value>
[SENSe]:REANalyze:SELect:FRAMe:STARt?
```

Arguments <value> :: <NR1> is an integer in the range from one to the number of fast frames.

[SENSe]:REANalyze:SELect:FRAMe:STARt:TIMestamp? (Query Only)

Queries the time stamp of the current Frame Start selection. This is the time stamp of the Frame Start frame of the acquisition start acquisition, if FastFrame is enabled.

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:SELect:FRAMe:STARt:TIMestamp?`

[SENSe]:REANalyze:SELect:FRAMe:STOP

Sets or queries the frame stop value in the currently selected set.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:REANalyze:SELect:FRAMe:STOP <value>
[SENSe]:REANalyze:SELect:FRAMe:STOP?
```

Related Commands [SENSe]:REANalyze:SELect:FRAMe:STOP:TIMestamp?

Arguments `<value> ::` `<NR1>` is an integer in the range from one to the number of fast frames.

[SENSe]:REANalyze:SELect:FRAMe:STOP:TIMestamp? (Query Only)

Queries the time stamp of the current frame stop selection. It is the time stamp of the frame Stop frame of the Acquisition Stop acquisition, if FastFrame is enabled.

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:SELect:FRAMe:STOP:TIMestamp?`

Related Commands [SENSe]:REANalyze:SELect:FRAMe:STOP

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:ALL (No Query Form)

Selects all spectra acquisitions and frames in the currently selected set.

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:ALL`

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:CURRent:ACQuisition? (Query Only)

Queries the acquisition record identifier (ID) of the currently selected acquisition.

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:CURRent:ACQuisition?`

Returns <ID Number> ::= <NR1> the record identifier number of the current acquisition.

Examples `[SENSE]:REANALYZE:SELECT:SPECTRA:ACQUISITION:CURRENT:ACQUISITION?` might return `848` indicating that the current acquisition record number is 848.

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:CURRent:TIMestamp? (Query Only)

Queries the time stamp of the currently selected acquisition.

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:CURRent:TIMestamp?`

Returns <time stamp> ::= <string> The current time stamp value.

Examples `[SENSE]:REANALYZE:SELECT:SPECTRA:ACQUISITION:CURRENT:TIMESTAMP?` might return "`10/10/2011 10:57:55:93`" indicating that the current time stamp recorded on October 10, 2011 was 10 hours, 57 minutes, and 55.93 seconds.

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:FIRSt? (Query Only)

Queries the first selectable acquisition start or stop value in the currently selected set.

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:FIRSt?`

Returns <ID Number> ::= <NR1> the record identifier number of the current acquisition.

Examples `[SENSE]:REANALYZE:SELECT:SPECTRA:ACQUISITION:FIRST?` might return `1` indicating that the first selectable acquisition start or stop value was 1.

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:FIRSt:TIMestamp? (Query Only)

Queries the time stamp of the first selectable acquisition history start or stop value in the currently selected acquisition set.

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:FIRSt:TIMestamp?`

Returns <time stamp> ::= <string> The first time stamp value.

Examples `[SENSE]:REANALYZE:SELECT:SPECTRA:ACQUISITION:FIRST:TIMESTAMP?` might return "`10/10/2011 10:57:41:58`" indicating that the first time stamp recorded on October 10, 2011 was 10 hours, 57 minutes, and 41.58 seconds.

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:LAST? (Query Only)

Queries the last selectable acquisition start or stop value in the currently selected set.

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:LAST?`

Returns <ID Number> ::= <NR1> the record identifier number of the current acquisition.

Examples `[SENSE]:REANALYZE:SELECT:SPECTRA:ACQUISITION:LAST?` might return `1` indicating that the last selectable acquisition start or stop value was 1.

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:LAST:TIMestamp? (Query Only)

Queries the time stamp of the last selectable acquisition start or stop value in the currently selected set.

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:LAST:TIMestamp?`

Returns <time stamp> ::= <string> The last time stamp value.

Examples `[SENSE]:REANALYZE:SELECT:SPECTRA:ACQUISITION:LAST:TIMESTAMP?` might return "`10/10/2011 10:57:41:58`" indicating that the last time stamp recorded on October 10, 2011 was 10 hours, 57 minutes, and 41.58 seconds.

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STARt

Sets or queries the acquisition start value in the currently selected set.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STARt <value>
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STARt?
```

Arguments <value> ::= <NR1> is an integer in the range from one to the number of acquisitions in the acquisition history.

Examples `[SENSE]:REANALYZE:SELECT:SPECTRA:ACQUISITION:START?` might return `1.0000000000` indicating the start value was 1.

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STARt:TIMestamp? (Query Only)

Queries the time stamp of the current acquisition start selection.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STARt:
TIMestamp?
```

Returns <time stamp> ::= <string> The start time stamp value.

Examples `[SENSE]:REANALYZE:SELECT:SPECTRA:ACQUISITION:START:TIMESTAMP?` might return "`10/10/2011 10:57:41:58`" indicating that the

time stamp of the acquisition start selection recorded on October 10, 2011 was 10 hours, 57 minutes, and 41.58 seconds.

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STOP

Sets or queries the acquisition stop value in the currently selected set.

Conditions Measurement views: All

Group Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STOP <value>
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STOP?
```

Arguments <value> ::= <NR1> is an integer in the range from one to the number of acquisitions in the acquisition history.

Examples `[SENSE]:REANALYZE:SELECT:SPECTRA:ACQUISITION:STOP?` might return `1.0000000000` indicating the stop value was 1.

[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STOP:TIMestamp? (Query Only)

Queries the time stamp of the current acquisition stop selection.

Conditions Measurement views: All

Group Sense commands

Syntax

```
[SENSe]:REANalyze:SELect:SPECtra:ACQuisition:STOP:TIMestamp?
```

Returns <time stamp> ::= <string> The stop time stamp value.

Examples `[SENSE]:REANALYZE:SELECT:SPECTRA:ACQUISITION:STOP:TIMESTAMP?` might return "`10/10/2011 10:57:41:58`" indicating that the time stamp of stop selection recorded on October 10, 2011 was 10 hours, 57 minutes, and 41.58 seconds.

[SENSe]:REANalyze:SPEed

Sets or queries the time delay value inserted between reanalysis of frames or acquisitions.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:REANalyze:SPEed <value>
[SENSe]:REANalyze:SPEed?
```

Arguments <value> :: <NR1> is an integer in the range of zero (for no delay) to four (for a maximum delay of about one second).

[SENSe]:REANalyze:STOP (No Query Form)

Stops the current reanalysis. The command will block until the current reanalysis has completed and stopped.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:REANalyze:STOP
```

[SENSe]:REANalyze:TIMestamp:DECimal

Sets or queries the number of values to the right of the decimal point to include in all time stamps.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:REANalyze:TIMestamp:DECimal <value>
[SENSe]:REANalyze:TIMestamp:DECimal?
```

Arguments `<value> ::` `<NR1>` is an integer in the range of one to nine.

[SENSe]:ROSCillator:SOURce

Sets or queries the frequency reference oscillator source.

Conditions Measurement views: All

Group Sense commands

Syntax

```
[SENSe]:ROSCillator:SOURce { INTernal | EXTernal }
[SENSe]:ROSCillator:SOURce?
```

Arguments `INTernal` specifies that the analyzer uses the 10 MHz internal oscillator.

`EXTernal` specifies that the analyzer uses the external reference signal. Use the Ref In connector on the rear panel to input the signal.

Selecting EXTernal initiates an attempt to lock the internal reference oscillator to the external reference signal. If the signal is not connected or is at an invalid frequency or amplitude, an error (2028, "External frequency reference signal not valid. Using internal reference) is returned. If not able to lock to the external reference, an error (2029, "Unable to lock to external frequency reference. Using internal reference.") is returned.

Examples `SENSE:ROSCILLATOR:SOURCE INTernal` uses the 10 MHz internal oscillator for the frequency reference.

[SENSe]:SEM:{BANDwidth|BWIDth}[:RESolution]

Sets or queries the resolution bandwidth (RBW) of the reference channel.

NOTE. *In Real-time operation this sets the RBW for the reference and all offset channels.*

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax

```
[SENSe]:SEM:{BANDwidth|BWIDth}[:RESolution] <value>
[SENSe]:SEM:{BANDwidth|BWIDth}[:RESolution]?
```

Arguments <value>::<NRf> specifies the resolution bandwidth.

Examples `[SENSE]:SEM:{BANDWIDTH|BWIDTH}[:RESOLUTION] 200kHZ` sets the resolution bandwidth to 200 kHz.

[SENSe]:SEM:{BANDwidth|BWIDth}:VIDeo

Sets or queries the video bandwidth (VBW) of the reference channel.

NOTE. *In Real-time operation this sets the RBW for the reference and all offset channels.*

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax

```
[SENSe]:SEM:{BANDwidth|BWIDth}:VIDeo <value>
[SENSe]:SEM:{BANDwidth|BWIDth}:VIDeo?
```

Arguments <value>::<NRf> specifies the video bandwidth.

Range: Current RBW/10^4 (1 Hz minimum) to current RBW.

Examples `[SENSE]:SEM:{BANDWIDTH|BWIDTH}:VIDEO 200kHZ` sets the video bandwidth to 200 kHz.

[SENSe]:SEM:{BANDwidth|BWIDth}:VIDeo:STATe

Sets or queries the state of the video bandwidth (VBW) of the reference channel.

NOTE. *In Real-time operation this sets the VBW state for the reference and all offset channels.*

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax
```
[SENSe]:SEM:{BANDwidth|BWIDth}:VIDeo:STATe { OFF | ON | 0 | 1 }
[SENSe]:SEM:{BANDwidth|BWIDth}:VIDeo:STATe?
```

Arguments `OFF` or `0` turns the video bandwidth filter off.

`ON` or `1` turns the video bandwidth filter on.

[SENSe]:SEM:CHANnel:FILTer

Sets or queries the reference Channel Filter.

NOTE. *In Real-time operation the channel filter is always forced to None.*

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax
```
[SENSe]:SEM:CHANnel:FILTer { RRC | NONE }
[SENSe]:SEM:CHANnel:FILTer?
```

Arguments `RRC` selects the Root-Raised-Cosine filter.

`NONE` uses no filter.

Examples `[SENSE]:SEM:CHANNEL:FILTER RRCosine` selects Root-Raised-Cosine for the reference Channel Filter.

[SENSe]:SEM:CHANnel:IBANdwidth

Sets or queries the reference channel integration bandwidth.

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax

```
[SENSe]:SEM:CHANnel:IBANdwidth <value>
[SENSe]:SEM:CHANnel:IBANdwidth?
```

Arguments <value>::<NRf> integration bandwidth

Returns

Examples `[SENSE]:SEM:CHANNEL:IBANDWIDTH?` might return `18.000000E+6` indicating that the integration bandwidth is 18 MHz.

[SENSe]:SEM:CHANnel:WIDTh

Sets or queries the reference Channel bandwidth.

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax

```
[SENSe]:SEM:CHANnel:WIDTh <value>
[SENSe]:SEM:CHANnel:WIDTh?
```

Arguments <value>::<NRf> specifies the filter bandwidth.

Examples `[SENSE]:SEM:CHANNEL:WIDTH 30kHZ` sets the filter bandwidth to 30 kHz.

[SENSe]:SEM:CHIPrate

Sets or queries the Chip rate.

Use the [SENSe]:SEM:CHANnel:FILTercommand to set the reference Channel Filter to RRC (Root-Raised Cosine).

NOTE. *Applies in Swept operation only.*

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax

```
[SENSe]:SEM:CHIPrate <value>
[SENSe]:SEM:CHIPrate?
```

Arguments <value>::<NRf> specifies the chip rate.

Examples `[SENSE]:SEM:CHIPRATE 3.84MHZ` sets the Chip rate toe 3.84 MHz.

[SENSe]:SEM:CLEar:RESults (No Query Form)

Clears the results in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax

```
[SENSe]:SEM:CLEar:RESults
```

Arguments None

Examples `[SENSE]:SEM:CLEAR:RESULTS` clears the results in the measurement.

[SENSe]:SEM:FREQuency

Sets or queries the measurement frequency.

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax

```
[SENSe]:SEM:FREQuency <value>
[SENSe]:SEM:FREQuency?
```

Arguments <value>::<NRf> specifies the center frequency.

Examples `[SENSE]:SEM:FREQUENCY 1.0GHZ` sets the center frequency to 1.0 GHz.

[SENSe]:SEM:FREQuency:SPAN? (Query Only)

Returns the frequency span.

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax `[SENSe]:SEM:FREQuency:SPAN?`

Arguments None

Returns <NRf> frequency span.

Examples `[SENSE]:SEM:FREQUENCY:SPAN?` might return `100.00000E+6` indicating that the frequency span is 100 MHz.

[SENSe]:SEM:MFILter

Sets or queries the shape of the filter.

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax
```
[SENSe]:SEM:MFILter { GAUSsian | RECTangular }
[SENSe]:SEM:MFILter?
```

Arguments `GAUSsian` sets the filter shape to Gaussian.

`RECTangular` sets the filter shape to Rectangular.

Examples `[SENSE]:SEM:MFILTER RECTangular` sets the filter shape to Rectangular.

[SENSe]:SEM:MODE:REALtime

Sets or queries the Real-time SEM measurement mode.

NOTE. *In real-time mode SEM is measured using real-time acquisition data.*

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax
```
[SENSe]:SEM:MODE:REALtime { OFF | ON | 0 | 1 }
[SENSe]:SEM:MODE:REALtime?
```

Arguments `ON` or `1` turns the SEM real-time measurement mode on.

`OFF` or `0` turns the SEM real-time measurement mode off.

[SENSe]:SEM:MTYPe

Sets or queries the measurement type.

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax
```
[SENSe]:SEM:MTYPe { TPOW | PSD | PEAK }
```

Arguments `TPOW` specifies the Total Power measurement.

`PSD` specifies the Peak Spectral Density measurement in dBM/Hz.

`PEAK` specifies a Peak power measurement.

Examples `[SENSE]:SEM:MTYPE PEAK` sets the measurement type to Peak.

[SENSe]:SEM:OFFSet:DETection

Sets or queries the detection settings for all offsets.

NOTE. *In Real-time operation the offset and reference detection settings are forced equal.*

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax
```
[SENSe]:SEM:OFFSet:DETection { AVERage | PEAK }
[SENSe]:SEM:OFFSet:DETection?
```

Arguments AVERage indicates the Offsets setting is Average (VRMS).

PEAK indicates the Offsets setting is the positive peak.

Examples [SENSE]:SEM:OFFSET:DETECTION AVER sets the detection settings for all offsets to Average RMS.

[SENSe]:SEM:OFFSet<x>:BANDwidth:FILTer? (Query Only)

Returns the bandwidth filter setting for specified row in the Offsets & Limits table. The filter setting is equal to the Resolution Bandwidth (RBW) value times the number of resolution bandwidths. The parameter <x> represents the row in the Offsets & Limits table.

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax
```
[SENSe]:SEM:OFFSet<x>:BANDwidth:FILTer?
```

Arguments None

Returns <value>::<NRf> is the filter bandwidth setting.

Examples [SENSe]:SEM:OFFSet3:BANDwidth:FILTer might return 30.0E+3 indicating that the filter bandwidth for row 3 to 30 kHz.

[SENSe]:SEM:OFFSet<x>:BANDwidth[:RESolution]

Sets or queries the resolution bandwidth for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table. See the [SENSe]:SEM:{BANDwidth|BWIDth}[:RESolution] command.

NOTE. *In Real-time operation the RBW values for each offset and the reference are set equal to the All Channels RBW setting.*

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax

```
[SENSe]:SEM:OFFSet<x>:BANDwidth[:RESolution] <value>
[SENSe]:SEM:OFFSet<x>:BANDwidth[:RESolution]?
```

Returns <value>::<NRf> specifies the resolution bandwidth.

Examples [SENSe]:SEM:OFFSet4:BANDwidth[:RESolution] 30kHz sets the resolution bandwidth in row 4 to 30 kHz.

[SENSe]:SEM:OFFSet<x>:BANDwidth:RESolution:COUNt

Sets or queries the count of Resolution Bandwidths (RBWs) for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table.

NOTE. *In Real-time operation the RBW count is always forced to 1.*

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax

```
[SENSe]:SEM:OFFSet<x>:BANDwidth:RESolution:COUNt <value>
[SENSe]:SEM:OFFSet<x>:BANDwidth:RESolution:COUNt?
```

Arguments <value> specifies the count number of RBWs.

Examples `[SENSe]:SEM:OFFSet1:BANDwidth:RESolution:COUNt 10` sets the count for row 1 to 10.

[SENSe]:SEM:OFFSet<x>:BANDwidth:VIDeo

Sets or queries the video bandwidth setting for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table. See the [SENSe]:SEM:{BANDwidth|BWIDth}:VIDeo command.

NOTE. *In Real-time operation the VBW values for each offset and the reference are set equal to the All Channels VBW setting.*

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax
```
[SENSe]:SEM:OFFSet<x>:BANDwidth:VIDeo <value>
[SENSe]:SEM:OFFSet<x>:BANDwidth:VIDeo?
```

Arguments <value>::<NRf> specifies the video bandwidth setting.

Examples `[SENSe]:SEM:OFFSet3:BANDwidth:VIDeo 300` sets the video bandwidth for row 3 to 300 Hz.

[SENSe]:SEM:OFFSet<x>:BANDwidth:VIDeo:STATe

Sets or queries the state of the video bandwidth for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table. See the [SENSe]:SEM:{BANDwidth|BWIDth}:VIDeo:STATe command.

NOTE. *In Real-time operation the VBW state values for each offset and the reference are set equal to the All Channels VBW state setting.*

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax

```
[SENSe]:SEM:OFFSet<x>:BANDwidth:VIDeo:STATe { OFF | ON | 0 | 1 }
[SENSe]:SEM:OFFSet<x>:BANDwidth:VIDeo:STATe?
```

Arguments

`OFF` or `0` turns the video bandwidth filter off.

`ON` or `1` turns the video bandwidth filter on.

Examples

`[SENSe]:SEM:OFFSet3:BANDwidth:VIDeo:STATe ON` turns on the video bandwidth filter for row 3 in the table.

[SENSe]:SEM:OFFSet<x>:FREQuency:STARt

Sets or queries the start frequency for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:OFFSet<x>:FREQuency:STARt <value>
[SENSe]:SEM:OFFSet<x>:FREQuency:STARt?
```

Arguments

<value>::<NRf> specifies the start frequency.

Examples

`[SENSe]:SEM:OFFSet1:FREQuency:STARt 2.876MHZ` sets the start frequency in row 1 to 2.876 MHz.

[SENSe]:SEM:OFFSet<x>:FREQuency:STOP

Sets or queries the stop frequency for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table.

Conditions

Measurement views: Spectral Emissions Mask

Group

Sense commands

Syntax

```
[SENSe]:SEM:OFFSet<x>:FREQuency:STOP <value>
[SENSe]:SEM:OFFSet<x>:FREQuency:STOP?
```

Arguments <value>::<NRf> specifies the stop frequency.

Examples `[SENSe]:SEM:OFFSet1:FREQuency:STOP 3.485MHZ` ssets the stop frequency in row 1 to 3.485 MHz.

[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:SAME

Sets or queries the state of the limits for the absolute start and stop settings for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table.

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax
```
[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:SAME { OFF | ON | 0 | 1 }
[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:SAME?
```

Arguments `OFF` or `0` indicates the start and stop limits are different.

`ON` or `1` indicates the state and stop limits are the same.

Examples `[SENSe]:SEM:OFFSet2:LIMIT:ABSolute:SAME ON` sets the absolute start and stop limits for row 2 to the same values.

[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:STARt

Sets or queries the absolute power start limit for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table.

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax
```
[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:STARt <value>
[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:STARt?
```

Arguments <value>::<NRf> specifies the power in dBm.

Examples `[SENSe]:SEM:OFFSet1:LIMIT:ABSolute:STARt –71.10` sets the absolute start power limit for row 1 to –71.10 dBm.

[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:STOP

Sets or queries the absolute power stop limit for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax
```
[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:STOP <value>
[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:STOP?
```

Arguments <value>::<NRf> specifies the power in dBm.

Examples `[SENSe]:SEM:OFFSet<x>:LIMIT:ABSolute:STOP –55.80` sets the absolute stop power limit for row 1 to –55.80 dBm.

[SENSe]:SEM:OFFSet<x>:LIMIT:MASK

Sets or queries the limit mask setting for the specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table.

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax
```
[SENSe]:SEM:OFFSet<x>:LIMIT:MASK { ABS | REL | AND | OR }
[SENSe]:SEM:OFFSet<x>:LIMIT:MASK?
```

Arguments `ABS` specifies the limit mask is set to Absolute.

`REL` specifies the limit mask is set to Relative

`AND` specifies the limit mask is set to Absolute AND Relative.

`OR` specifies the limit mask is set to Absolute OR Relative.

Examples `[SENSe]:SEM:OFFSet3:LIMIT:MASK REL` sets the limit mask for row 3 to Relative.

[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:SAME

Sets or queries the state of the limits for the relative start and stop settings for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table.

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax
```
[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:SAME { OFF | ON | 0 | 1 }
[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:SAME?
```

Arguments `OFF` or `0` indicates the start and stop limits are different.

`ON` or `1` indicates the state and stop limits are the same.

Examples `[SENSe]:SEM:OFFSet2:LIMIT:RELative:SAME ON` sets the relative start and stop limits for row 2 to the same values.

[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:STARt

Sets or queries the relative power start limit for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table.

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax
```
[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:STARt <value>
[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:STARt?
```

Arguments <value>::<NRf> specifies the power in dB.

Examples `[SENSe]:SEM:OFFSet1:LIMIT:RELative:STARt -35.5` sets the relative start power limit for row 1 to –35.5 dB.

[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:STOP

Sets or queries the relative power stop limit for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table.

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax
```
[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:STOP <value>
[SENSe]:SEM:OFFSet<x>:LIMIT:RELative:STOP?
```

Arguments <value>::<NRf> specifies the power in dB.

Examples `[SENSe]:SEM:OFFSet1:LIMIT:RELative:STOP -35.5` sets the relative stop power limit for row 1 to 35.5 dB.

[SENSe]:SEM:OFFSet<x>:LIMIT:SIDE

Sets or queries the side of the offset limit for specified row in the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table.

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax
```
[SENSe]:SEM:OFFSet<x>:LIMIT:SIDE { LEFT | RIGHt | BOTH}
[SENSe]:SEM:OFFSet<x>:LIMIT:SIDE?
```

Arguments `LEFT` specifies the left offset limit.

`RIGHt` specifies the right offset limit.

`BOTH` specifies both sides of the offset limit.

Examples `[SENSe]:SEM:OFFSet1:LIMIT:SIDE BOTH` specifies the measurement uses both sides of the offset in row 1.

[SENSe]:SEM:OFFSet<x>:STATe (No Query Form)

Determines whether to display the specified row of the Offsets & Limits table. The parameter <x> represents the row in the Offsets & Limits table.

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax `[SENSe]:SEM:OFFSet<x>:STATe { OFF | ON | 0 | 1 }`

Arguments `OFF` or `0` turns off the display of the row in the measurement view.

`ON` or `1` displays the row in the measurement view.

Examples [SENSe]:SEM:OFFSet2:STATe OFF the display of row 2 in the measurement view.

[SENSe]:SEM:POINts:COUNt

Sets or queries the maximum number of trace points per range.

NOTE. *In Swept operation this is the number of trace points per range. In Real-time operation it is the number of trace points in the entire trace.*

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax
```
[SENSe]:SEM:POINts:COUNt <value>
[SENSe]:SEM:POINts:COUNt?
```

Arguments None

Returns P<NR1> number of points per range.

Examples `[SENSE]:SEM:POINTS:COUNT`? might return `P8001` indicating a maximum of 8001 trace points.

[SENSe]:SEM:PRESet (No Query Form)

Defines the standard protocol settings for the Spectral Emissions Mask measurement.

NOTE. *This is an obsolete command.*

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax
```
[SENSe]:SEM:PRESet { “3GPP Downlink” | “WLAN 802.11g” |
“WiMax 802.16 (ETSI)” }
```

Arguments “`3GPP Downlink`” specifies the 3GPP Downlink standard.

“`WLAN 802.11g`” specifies the IEEE 802.11 standard for WLAN communications.

“`WiMax 802.16 (ETSI)`” specifies the WiMAX 802.16 (ETSI) standard.

Quotes must be included with the arguments.

Examples `[SENSE]:SEM:PRESET WLAN (802.11g)` presets the SEM measurement to the IEEE 802.11 standard for WLAN communications.

[SENSe]:SEM:REFerence:AUTO

Determines whether to automatically update the Power Reference in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax
```
[SENSe]:SEM:REFerence:AUTO { OFF | ON | 0 | 1 }
[SENSe]:SEM:REFerence:AUTO?
```

Arguments `OFF` or `0` turns off the automatic update.

`ON` or `1` automatically updates the reference power.

Examples `[SENSE]:SEM:REFERENCE:AUTO ON` automatically updates the reference power in the measurement.

[SENSe]:SEM:REFerence:DETection

Sets or returns the reference channel detection setting.

NOTE. *In Real-time operation the offset and reference detection settings are forced equal.*

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax
```
[SENSe]:SEM:REFerence:DETection { AVERage | PEAK }
[SENSe]:SEM:REFerence:DETection?
```

Arguments `AVERage` indicates the reference channel detection setting is Average (VRMS).

`PEAK` indicates the reference channel detection setting is the positive peak.

Examples `[SENSE]:SEM:REFERENCE:DETECTION AVER` sets the reference channel detection setting to Average RMS.

[SENSe]:SEM:REFerence[:MAGNitude]

Sets or queries the reference power magnitude.

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax
```
[SENSe]:SEM:REFerence[:MAGNitude] <Value>
[SENSe]:SEM:REFerence[:MAGNitude]?
```

Arguments <value>::<NRf> specifies the magnitude in dBm.

Examples `[SENSE]:SEM:REFERENCE[:MAGNITUDE]?` might return a magnitude of –51.057.

[SENSe]:SEM:RRCRolloff

Sets or queries the filter parameter (roll-off ratio) for the Root Raised Cosine filter.

NOTE. *Applies in Swept operation only.*

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax
```
[SENSe]:SEM:RRCRolloff <value>
[SENSe]:SEM:RRCRolloff?
```

Arguments <value>::<NRf> specifies the filter parameter.

Range: 0.001 to 1, 0.001 step.

Examples `[SENSE]:SEM:RRCROLLOFF 0.22` sets the filter parameter to 0.22.

[SENSe]:SEM:STEP:AUTO

Determines whether to automatically update the center frequency increment/decrement step size in the Spectral Emissions Mask measurement. The Step value controls the magnitude of each step when using the front panel controls or arrow keys on the keyboard.

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax
```
[SENSe]:SEM:STEP:AUTO { OFF | ON | 0 | 1 }
[SENSe]:SEM:STEP:AUTO?
```

Related Commands [SENSe]:SEM:STEP[:MAGNitude]

Arguments OFF or 0 turns off the automatic update.

ON or 1 automatically updates center frequency.

Examples [SENSE]:SEM:STEP:AUTO ON automatically updates the center frequency.

[SENSe]:SEM:STEP[:MAGNitude]

Sets or queries the step magnitude for the center frequency in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax

```
[SENSe]:SEM:STEP[:MAGNitude] <value>
[SENSe]:SEM:STEP[:MAGNitude]?
```

Related Commands [SENSe]:SEM:STEP:AUTO

Arguments <value>::<NRf> specifies the step setting.

Examples [SENSE]:SEM:STEP[:MAGNITUDE] 4MHZ sets the step magnitude to 4 MHz.

[SENSe]:SEM:WIDTh:DEFinition

Sets or queries the offset definition of the measurement type in the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Sense commands

Syntax

```
[SENSe]:SEM:WIDTh:DEFinition {C2C | C2E | E2C | E2E}
[SENSe]:SEM:WIDTh:DEFinition?
```

Arguments

`C2C` defines the offset from the reference frequency center to the offset center.

`C2E` defines the offset from the reference frequency center to the offset edge.

`E2C` defines the offset from the reference frequency edge to the offset center.

`E2E` defines the offset from the reference frequency edge to the offset edge.

Examples

`[SENSE]:SEM:WIDTH:DEFINITION` C2C specifies the offset limit from the center of the reference frequency to the center of the offset.

[SENSe]:SGRam:{BANDwidth|BWIDth}:RESolution

Sets or queries the resolution bandwidth (RBW). Programming a specified RBW sets [SENSe]:SGRam{BANDwidth|BWIDth}:RESolution:AUTO OFF.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam:{BANDwidth|BWIDth}:RESolution <value>
[SENSe]:SGRam:{BANDwidth|BWIDth}:RESolution?
```

Related Commands

[SENSe]:SGRam:{BANDwidth|BWIDth}[:RESolution]:AUTO

Arguments

`<value> ::  <NRf>` specifies the RBW. Range: 1 Hz to 10 MHz.

Examples

`SENSE:SGRAM:BANDWIDTH:RESOLUTION 200kHz` sets the RBW to 200 kHz.

[SENSe]:SGRam:{BANDwidth|BWIDth}[:RESolution]:ACTual? (Query Only)

Queries the actual resolution bandwidth (RBW) in the spectrogram.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam:{BANDwidth|BWIDth}[:RESolution]:ACTual?
```

Arguments None

Returns <NRf> The actual RBW in Hz.

Examples SENSE:SGRAM:BANDWIDTH:RESOLUTION:ACTUAL? might return 299.624E+3, indicating that the actual RBW is 299.624 kHz.

[SENSe]:SGRam:{BANDwidth|BWIDth}[:RESolution]:AUTO

Determines whether to set the resolution bandwidth (RBW) automatically or manually in the spectrogram.

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax
```
[SENSe]:SGRam:{BANDwidth|BWIDth}[:RESolution]:AUTO { OFF | ON | 0 | 1 }
[SENSe]:SGRam:{BANDwidth|BWIDth}[:RESolution]:AUTO?
```

Arguments OFF or 0 specifies that the resolution bandwidth is set manually using the[SENSe]:DPX:{BANDwidth|BWIDth}:ACTual? command.

ON or 1 specifies that the resolution bandwidth is set automatically.

Examples SENSE:SGRAM:BANDWIDTH:RESOLUTION:AUTO ON sets the resolution bandwidth automatically.

[SENSe]:SGRam:{BANDwidth|BWIDth}[:RESolution]:MODE

Determines whether to enable or disable the RBW processing in the spectrogram.

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax
```
[SENSe]:SGRam:{BANDwidth|BWIDth}[:RESolution]:MODE { OFF | ON | 0 | 1 }
[SENSe]:SGRam:{BANDwidth|BWIDth}[:RESolution]:MODE?
```

Arguments

`OFF` or `0` disables the RBW processing. You can select the FFT window using the[SENSe]:SGRam:FFT:WINDow command.

`ON` or `1` enables the RBW processing. Refer to the [SENSe]:SGRam:{BANDwidth|BWIDth}:RESolutioncommand to set the RBW.

Examples

`SENSE:SGRAM:BANDWIDTH:RESOLUTION:MODE ON` enables the RBW processing.

[SENSe]:SGRam:{BANDwidth|BWIDth}:VIDeo

Sets or queries the video bandwidth (VBW). Programming a specified VBW sets [SENSe]:SGRam:{BANDwidth|BWIDth}:VIDeo:STATe OFF.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam:{BANDwidth|BWIDth}:VIDeo <value>
[SENSe]:SGRam:{BANDwidth|BWIDth}:VIDeo?
```

Related Commands

[SENSe]:SGRam:{BANDwidth|BWIDth}:VIDeo:STATe

Arguments

`<value> ::= <NRf>` specifies the VBW.
Range: Current RBW/10^4 (1 Hz minimum) to Current RBW.

Examples

`SENSE:SGRAM:BANDWIDTH:VIDEO 200kHz` sets the VBW to 200 kHz.

[SENSe]:SGRam:{BANDwidth|BWIDth}:VIDeo:STATe

Determines whether to enable or disable the video bandwidth (VBW) in the spectrogram measurement.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam:{BANDwidth|BWIDth}:VIDeo:STATe { OFF | ON |
0 | 1 }
```

```
[SENSe]:SGRam:{BANDwidth|BWIDth}:VIDeo:STATe?
```

Arguments `OFF` or `0` disables the VBW.

`ON` or `1` enables the VBW.

Examples `SENSE:SGRAM:BANDWIDTH:VIDEO:STATE ON` enables the VBW.

[SENSe]:SGRam:COLor

Sets or queries the color palette of three-dimensional graphs.

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax

```
[SENSe]:SGRam:COLor { RED | GREen | BLUe | CYAN | BCYan|
YELLow | MAGenta | GRAY | TEMPerature | SPECtral }
[SENSe]:SGRam:COLor?
```

Arguments The following table lists the arguments.

Table 2-36: Color palette for spectrogram

Argument	Palette
RED	Red
GREen	Green
BLUe	Blue
CYAN	Cyan
BCYan	Binary cyan
YELLow	Yellow
MAGenta	Magenta
GRAY	Gray
TEMPerature	Temperature
SPECtral	Spectral

Examples `SENSE:SGRAM:COLOR TEMPerature` selects the temperature color palette.

[SENSe]:SGRam:COLor:MAXimum

Sets or queries the maximum value of the color axis in the spectrogram.

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax
```
[SENSe]:SGRam:COLor:MAXimum <value>
[SENSe]:SGRam:COLor:MAXimum?
```

Related Commands [SENSe]:SGRam:COLor:MINimum

Arguments `<value> ::= <NRf>` specifies the maximum value of the color axis.
Range: -100 to +100 dBm.

Examples `SENSE:SGRAM:COLOR:MAXIMUM 10` sets the maximum value of the color axis to 10 dBm.

[SENSe]:SGRam:COLor:MINimum

Sets or queries the minimum value of the color axis in the spectrogram.

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax
```
[SENSe]:SGRam:COLor:MINimum <value>
[SENSe]:SGRam:COLor:MINimum?
```

Related Commands [SENSe]:SGRam:COLor:MAXimum

Arguments `<value> ::= <NRf>` specifies the minimum value of the color axis.
Range: -100 to +100 dBm.

Examples `SENSE:SGRAM:COLOR:MINIMUM 10` sets the minimum value of the color axis to 10 dBm.

[SENSe]:SGRam:FFT:WINDow

Sets or queries the FFT window in the spectrogram.This command is equivalent to [SENSe]:SGRam:FILTer[:SHAPe].

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax

```
[SENSe]:SGRam:FFT:WINDow { KAISer | MIL6db | CISPr | BH4B |
UNIForm | FLATtop | HANNing }
[SENSe]:SGRam:FFT:WINDow?
```

Arguments

`KAISer` selects the Kaiser (RBW) window.

`MIL6db` selects the -6 dB RBW (MIL) window.

`CISPr` selects the CISPR window.

`FLATtop` selects the flat-top window.

`HANNing` selects the Hanning window.

`BH4B` selects the Blackman-Harris 4B type window.

`UNIForm` selects the uniform window.

Examples `SENSE:SGRAM:FFT:WINDOW HANNing` selects the Hanning window.

[SENSe]:SGRam:FILTer[:SHAPe]

Sets or queries the filter shape in the spectrogram. This command is equivalent to [SENSe]:SGRam:FFT:WINDow.

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax

```
[SENSe]:SGRam:FILTer[:SHAPe] { KAISer | MIL6db | CISPr |
BH4B | UNIForm | FLATtop | HANNing }
[SENSe]:SGRam:FILTer[:SHAPe]?
```

Arguments

`KAISer` selects the Kaiser (RBW) window.

`MIL6db` selects the -6 dB RBW (MIL) window.

`CISPr` selects the CISPR window.

`FLATtop` selects the flat-top window.

`HANNing` selects the Hanning window.

`BH4B` selects the Blackman-Harris 4B type window.

`UNIForm` selects the uniform window.

Examples

`SENSE:SGRAM:FILTER:SHAPE HANNing` selects the Hanning window.

[SENSe]:SGRam:FREQuency:CENTer

Sets or queries the center frequency in the spectrogram.

NOTE. *The center, start and stop frequencies are set interlocking each other with the following relationships: (start frequency) = (center frequency) - (span)/2 and (stop frequency) = (center frequency) + (span)/2.*

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam:FREQuency:CENTer <value>
[SENSe]:SGRam:FREQuency:CENTer?
```

Related Commands

[SENSe]:SGRam:FREQuency:STARt, [SENSe]:SGRam:FREQuency:STOP

Arguments

`<value> ::  <NRf>` specifies the center frequency.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `SENSE:SGRAM:FREQUENCY:CENTER 7.5GHz` sets the center frequency to 7.5 GHz.

[SENSe]:SGRam:FREQuency:SPAN

Sets or queries the frequency span in the spectrogram.

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax
```
[SENSe]:SGRam:FREQuency:SPAN <value>
[SENSe]:SGRam:FREQuency:SPAN?
```

Arguments `<value> ::` `<NRf>` specifies the frequency span.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `SENSE:SGRAM:FREQUENCY:SPAN 20MHz` sets the span to 20 MHz.

[SENSe]:SGRam:FREQuency:SPAN:BANDwidth[:RESolution]:RATio

Sets or queries the ratio of span to RBW (Resolution Bandwidth) in the spectrogram. This command is valid when [SENSe]:SGRam:{BANDwidth|BWIDth}[:RESolution]:AUTO is set to On.

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax
```
[SENSe]:SGRam:FREQuency:SPAN:BANDwidth[:RESolution]:RATio
<value>
[SENSe]:SGRam:FREQuency:SPAN:BANDwidth[:RESolution]:RATio?
```

Arguments `<value> ::  <NRf>` specifies the ratio of span to RBW. Range: 20 to 1000.

Programming a specified ratio sets the RBW (= span/ratio), which is rounded down to the nearest valid value.

Examples `SENSE:SGRAM:FREQUENCY:SPAN:BANDWIDTH:RESOLUTION:RATIO 200` sets the ratio to 200, setting the RBW to 200 kHz for the span of 40 MHz.

[SENSe]:SGRam:FREQuency:SPAN:MAXimum (No Query Form)

Sets the frequency range to the maximum real-time span in the spectrogram.

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax `[SENSe]:SGRam:FREQuency:SPAN:MAXimum`

Arguments None

Examples `SENSE:SGRAM:FREQUENCY:SPAN:MAXIMUM` sets the frequency range to the maximum real-time span.

[SENSe]:SGRam:FREQuency:STARt

Sets or queries the measurement start frequency (left edge of the graph) in the spectrogram.

The center, start and stop frequencies are set interlocking each other. Refer to the [SENSe]:SGRam:FREQuency:CENTer command.

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax
```
[SENSe]:SGRam:FREQuency:STARt <value>
[SENSe]:SGRam:FREQuency:STARt?
```

Related Commands [SENSe]:SGRam:FREQuency:STOP

Arguments `<value> ::` `<NRf>` is the measurement start frequency.
Range: (center frequency) ± (span)/2.

Examples `SENSE:SGRAM:FREQUENCY:START 6.95GHz` sets the start frequency to 6.95 GHz.

[SENSe]:SGRam:FREQuency:STEP

Sets or queries the frequency step size. Programming a specified step size sets [SENSe]:SGRam:FREQuency:STEP:AUTO OFF.

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax
```
[SENSe]:SGRam:FREQuency:STEP <value>
[SENSe]:SGRam:FREQuency:STEP?
```

Related Commands [SENSe]:SGRam:FREQuency:STEP:AUTO

Arguments `<value> ::` `<NRf>` specifies the frequency step size.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `SENSE:SGRAM:FREQUENCY:STEP 1kHz` sets the frequency step size to 1 kHz.

[SENSe]:SGRam:FREQuency:STEP:AUTO

Determines whether to set the frequency step size automatically or manually in the spectrogram.

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax

```
[SENSe]:SGRam:FREQuency:STEP:AUTO { OFF | ON | 0 | 1 }
[SENSe]:SGRam:FREQuency:STEP:AUTO?
```

Arguments

OFF or 0 specifies that the frequency step size is set manually using the [SENSe]:SGRam:FREQuency:STEP command.

ON or 1 specifies that the frequency step size is set automatically.

Examples

SENSE:SGRAM:FREQUENCY:STEP:AUTO ON specifies that the frequency step size is set automatically.

[SENSe]:SGRam:FREQuency:STOP

Sets or queries the measurement stop frequency (right edge of the graph) in the spectrogram.

The center, start and stop frequencies are set interlocking each other. Refer to the [SENSe]:SGRam:FREQuency:CENTer command.

Conditions

Measurement views: Spectrogram

Group

Sense commands

Syntax

```
[SENSe]:SGRam:FREQuency:STOP <value>
[SENSe]:SGRam:FREQuency:STOP?
```

Related Commands

[SENSe]:SGRam:FREQuency:STARt

Arguments

<value> :: <NRf> is the measurement start frequency.
Range: (center frequency) ± (span)/2.

Examples

SENSE:SGRAM:FREQUENCY:STOP 7.05GHz sets the stop frequency to 7.05 GHz.

[SENSe]:SGRam:TIME[:SCALe]:MODe

Sets or queries the vertical mode.

Conditions

Measurement views: Spectrogram

Group Sense commands

Syntax
```
[SENSe]:SGRam:TIME[:SCALe]:MODe {NORMal | MONitor}
[SENSe]:SGRam:TIME[:SCALe]:MODe?
```

Examples `SENSE:SGRAM:TIME:SCALE:MODE MONITOR` set the instrument to Spectrum Monitor mode.

[SENSe]:SGRam:TIME[:SCALe]:OVERlap:PERCent

Sets or queries the Overlap in percent. Note that changing this value may change the time per divisionDISPlay:SGRam:TIME:SCALe:PER:DIVision value.

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax
```
[SENSe]:SGRam:TIME[:SCALe]:OVERlap:PERCent
[SENSe]:SGRam:TIME[:SCALe]:OVERlap:PERCent
```

Arguments `<value> ::  <NRf>` specifies the overlap in percent.

Examples `SENSE:SGRAM:TIME:SCALE:OVERLAP:PERCENT 1.0E1` sets the overlap to 10%.

[SENSe]:SGRam:TIME[:SCALe]:PER:DIVision

Sets or queries the vertical Time/division value. Note that changing this value may change the DISPlay:SGRam:TIME:OVERlap:PERCent value.

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax
```
[SENSe]:SGRam:TIME[:SCALe]:PER:DIVision <value>
[SENSe]:SGRam:TIME[:SCALe]:PER:DIVision?
```

Related Commands DISPlay:SGRam:TIME:OVERlap:PERCent

Arguments `<value> ::` `<NRf>` specifies the time per division in seconds.

Examples `SENSE:SGRAM:TIME:SCALE:PER:DIVISION?` might return `1.1160000000E-3`.

[SENSe]:SGRam:TIME[:SCALe]:PER:UPDate:MINutes

Sets or queries the Spectrum Monitor mode Time/Update value for minutes.

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax
```
[SENSe]:SGRam:TIME[:SCALe]:PER:UPDate:MINutes <value>
[SENSe]:SGRam:TIME[:SCALe]:PER:UPDate:MINutes?
```

Arguments `<value> ::` `<NRf>` sets the time/update value in minutes.

Examples `SENSE:SGRAM:TIME:SCALE:PER:UPDATE:MINUTES 1E1` sets the Time/update minutes value to 10.

[SENSe]:SGRam:TIME[:SCALe]:PER:UPDate:SEConds

Sets or queries the Spectrum Monitor mode Time/Update value for seconds.

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax
```
[SENSe]:SGRam:TIME[:SCALe]:PER:UPDate:SEConds <value>
[SENSe]:SGRam:TIME[:SCALe]:PER:UPDate:SEConds?
```

Arguments `<value> ::` `<NRf>` sets the time/update value in minutes.

Examples `SENSE:SGRAM:TIME:SCALE:PER:UPDATE:SECONDS 1E1` sets the Time/update seconds value to 10.

[SENSe]:SGRam:TIME[:SCALe]:SPECtrums:PERLine? (Query Only)

Queries the Spectrums/Line value.

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax `[SENSe]:SGRam:TIME[:SCALe]:SPECtrums:PERLine?`

Examples `SENSE:SGRAM:TIME:SCALE:SPECTRUMS:PERLINE?` might return a 1.0000000000 as a response.

[SENSe]:SGRam:TIME[:SCALe]:STARt:DIVisions

Sets or queries the Vertical offset or Position in divisions.

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax
```
[SENSe]:SGRam:TIME[:SCALe]:STARt:DIVisions <value>
[SENSe]:SGRam:TIME[:SCALe]:STARt:DIVisions?
```

Arguments `<value> ::= <NRf>` sets the vertical offset value in divisions. Range is -130,000 to +130,000.

Examples `SENSE:SGRAM:TIME:OFFSET:DIVISIONS 10` sets the vertical offset to 10 divisions

[SENSe]:SGRam:WATerfall:DIRection

Sets or queries the direction of the Waterfall (3-D spectrogram) display.

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax
```
[SENSe]:SGRam:WATerfall:DIRection { NW | NE }
[SENSe]:SGRam:WATerfall:DIRection?
```

Arguments `NW` sets the display direction to northwest.

`NE` sets the display direction to northeast.

Examples `SENSE:SGRAM:WATERFALL:DIRECTION NW` sets the waterfall display direction to northwest.

[SENSe]:SGRam:WATerfall:ENABle

Sets or queries the Waterfall (3-D spectrogram) display.

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax
```
[SENSe]:SGRam:WATerfall:ENABle { OFF | ON | 0 | 1 }
[SENSe]:SGRam:WATerfall:ENABle?
```

Arguments `OFF` or `0` disables the Waterfall display.

`ON` or `1` enables the Waterfall display.

Examples `SENSE:SGRAM:WATERFALL:ENABLE ON` enables the 3-D Waterfall display.

[SENSe]:SGRam:WATerfall:Y:AUTO (No Query Form)

Rescales the height axis automatically to fit the waveform to the screen in the 3-D Waterfall display.

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax `[SENSe]:SGRam:WATerfall:Y:AUTO`

Arguments None

Examples `SENSE:SGRAM:WATERFALL:Y:AUTO` rescales the height scale automatically to fit the waveform to the 3-D Waterfall display.

[SENSe]:SGRam:WATerfall:Y:OFFSet

Sets or queries the height offset (the value at the top edge of the vertical axis) of the 3-D Waterfall display.

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax
```
[SENSe]:SGRam:WATerfall:Y:OFFSet <value>
[SENSe]:SGRam:WATerfall:Y:OFFSet?
```

Arguments `<value> ::= <NRf>` specifies the height (vertical) offset. Range: -270 to +150 dBm.

Examples `SENSE:SGRAM:WATERFALL:Y:OFFSET 12.5` sets the offset to 12.5 dB in the 3-D Waterfall display.

[SENSe]:SGRam:WATerfall:Y:RESet (No Query Form)

Resets the height scale of the 3-D Waterfall display to the default values: Height position= 0 dB and Height scale = 100 dB

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax `[SENSe]:SGRam:WATerfall:Y:RESet`

Arguments None

Examples `SENSE:SGRAM:WATERFALL:Y:AUTO` rescales the height scale automatically to fit the waveform to the 3-D Waterfall display.

[SENSe]:SGRam:WATerfall:Y[:SCALe]

Sets or queries the height (vertical) range of the Waterfall (3-D spectrogram) display.

Conditions Measurement views: Spectrogram

Group Sense commands

Syntax

```
[SENSe]:SGRam:WATerfall:Y[:SCALe] <value>
[SENSe]:SGRam:WATerfall:Y[:SCALe]?
```

Arguments `<value> ::= <NRf>` specifies the vertical range. Range: 0.1 to 200 dB. The amplitude unit is set by the [SENSe]:POWer:UNITs command.

You can omit the unit in the argument. When you include the unit in the argument, only dB is available. For the amplitude units Watts, Volts, and Amps, omit the unit in the argument.

Examples `SENSE:SGRAM:WATERFALL:Y:SCALE 100dB` sets the height range to 100 dB in the 3-D Waterfall display.

[SENSe]:SIGNalpath:PRIority

Sets or queries the signal path priority for the measurements.

Conditions Measurement views: All

Group Sense commands

Syntax

```
[SENSe]:SIGNalpath:PRIority { RF | LF }
[SENSe]:SIGNalpath:PRIority?
```

Arguments

`RF` sets the signal path to high frequency range.

`LF` sets the signal path to low frequency range.

Examples

`SENSE:SIGNALPATH:PRIORITY` sets the low frequency signal path when measurements are completely within LF frequency range.

[SENSe]:SPECtrum:{BANDwidth|BWIDth}[:RESolution]

Sets or queries the resolution bandwidth (RBW). Programming a specified RBW sets [SENSe]:SPECtrum:{BANDwidth|BWIDth}[:RESolution]:AUTO OFF.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum:{BANDwidth|BWIDth}[:RESolution] <value>
[SENSe]:SPECtrum:{BANDwidth|BWIDth}[:RESolution]?
```

Related Commands

[SENSe]:SPECtrum:{BANDwidth|BWIDth}[:RESolution]:AUTO

Arguments

`<value> ::  <NRf>` specifies the RBW. Range: 1 Hz to 10 MHz.

Examples

`SENSE:SPECTRUM:BANDWIDTH:RESOLUTION 200kHz` sets the RBW to 200 kHz.

[SENSe]:SPECtrum:{BANDwidth|BWIDth}[:RESolution]:ACTual? (Query Only)

Queries the actual resolution bandwidth (RBW) in the spectrum measurement.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum:{BANDwidth|BWIDth}[:RESolution]:ACTual?
```

Arguments

None

Returns <NRf> The actual RBW in Hz.

Examples SENSE:SPECTRUM:BANDWIDTH:RESOLUTION:ACTUAL? might return 299.624E+3, indicating that the actual RBW is 299.624 kHz.

[SENSe]:SPECtrum:{BANDwidth|BWIDth}[:RESolution]:AUTO

Determines whether to set the resolution bandwidth (RBW) automatically or manually in the spectrum measurement.

Conditions Measurement views: Spectrum

Group Sense commands

Syntax

```
[SENSe]:SPECtrum:{BANDwidth|BWIDth}[:RESolution]:AUTO { OFF
| ON | 0 | 1 }
[SENSe]:SPECtrum:{BANDwidth|BWIDth}[:RESolution]:AUTO?
```

Arguments OFF or 0 specifies that the resolution bandwidth is set manually using the [SENSe]:SPECtrum:{BANDwidth|BWIDth}[:RESolution] command.

ON or 1 specifies that the resolution bandwidth is set automatically.

Examples SENSE:SPECTRUM:BANDWIDTH:RESOLUTION:AUTO ON sets the resolution bandwidth automatically.

[SENSe]:SPECtrum:{BANDwidth|BWIDth}[:RESolution]:MODE

Determines whether to enable or disable the RBW process.

NOTE. *This is an obsolete command.*

Conditions Measurement views: Spectrum

Group Sense commands

Syntax

```
[SENSe]:SPECtrum:{BANDwidth|BWIDth}[:RESolution]:MODE { OFF
| ON | 0 | 1 }
[SENSe]:SPECtrum:{BANDwidth|BWIDth}[:RESolution]:MODE?
```

Arguments

OFF or 0 disables the RBW process. You can select the FFT window using the [SENSe]:SPECtrum:FFT:WINDow command.

ON or 1 enables the RBW process. Refer to the [SENSe]:SPECtrum:{BANDwidth|BWIDth}[:RESolution] command to set the RBW.

Examples

SENSE:SPECTRUM:BANDWIDTH:RESOLUTION:MODE ON enables the RBW process.

[SENSe]:SPECtrum:{BANDwidth|BWIDth}:VIDeo

Sets or queries the video bandwidth (VBW). Programming a specified VBW sets [SENSe]:SPECtrum{BANDwidth|BWIDth}:VIDeo:STATe OFF.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum:{BANDwidth|BWIDth}:VIDeo <value>
[SENSe]:SPECtrum:{BANDwidth|BWIDth}:VIDeo?
```

Arguments

<value> :: <NRf> specifies the VBW.
Range: Current RBW/10^4 (1 Hz minimum) to Current RBW.

Examples

SENSE:SPECTRUM:BANDWIDTH:VIDEO 200kHz sets the VBW to 200 kHz.

[SENSe]:SPECtrum:{BANDwidth|BWIDth}:VIDeo:STATe

Determines whether to enable or disable the video bandwidth (VBW) in the spectrum measurement.

Conditions

Measurement views: Spectrum

Group

Sense commands

Syntax

```
[SENSe]:SPECtrum:{BANDwidth|BWIDth}:VIDeo:STATe { OFF | ON
| 0 | 1 }
[SENSe]:SPECtrum:{BANDwidth|BWIDth}:VIDeo:STATe?
```

Arguments `OFF` or `0` disables the VBW.

`ON` or `1` enables the VBW.

Examples `SENSE:SPECTRUM:BANDWIDTH:VIDEO:STATE ON` enables the VBW.

[SENSe]:SPECtrum:CLEar:RESults (No Query Form)

Restarts multi-trace functions (Average and Max/Min Hold).

Conditions Measurement views: Spectrum

Group Sense commands

Syntax `[SENSe]:SPECtrum:CLEar:RESults`

Arguments None

Examples `SENSE:SPECTRUM:CLEAR:RESULTS` restarts multi-trace functions.

[SENSe]:SPECtrum:FFT:WINDow

Sets or queries the FFT window in the spectrum measurement.This command is equivalent to [SENSe]:SPECtrum:FILTer[:SHAPe].

Conditions Measurement views: Spectrum

Group Sense commands

Syntax
```
[SENSe]:SPECtrum:FFT:WINDow { KAISer | MIL6db | CISPr | BH4B
| UNIForm | FLATtop | HANNing }
[SENSe]:SPECtrum:FFT:WINDow?
```

Arguments `KAISer` selects the Kaiser (RBW) window.

`MIL6db` selects the -6 dB RBW (MIL) window.

`CISPr` selects the CISPR window.

`BH4B` selects the Blackman-Harris 4B type window.

UNIForm selects the uniform window.

FLATtop selects the flat-top window.

HANNing selects the Hanning window.

Examples SENSE:SPECTRUM:FFT:WINDOW HANNing selects the Hanning window.

[SENSe]:SPECtrum:FILTer[:SHAPe]

Sets or queries the filter shape in the spectrum measurement. This command is equivalent to [SENSe]:SPECtrum:FFT:WINDow.

Conditions Measurement views: Spectrum

Group Sense commands

Syntax
```
[SENSe]:SPECtrum:FILTer[:SHAPe] { KAISer | MIL6db | CISPr | BH4B | UNIForm | FLATtop | HANNing }
[SENSe]:SPECtrum:FILTer[:SHAPe]?
```

Arguments KAISer selects the Kaiser (RBW) window.

MIL6db selects the -6 dB RBW (MIL) window.

CISPr selects the CISPR window.

BH4B selects the Blackman-Harris 4B type window.

UNIForm selects the uniform window.

FLATtop selects the flat-top window.

HANNing selects the Hanning window.

Examples SENSE:SPECTRUM:FILTER:SHAPE HANNing selects the Hanning window.

[SENSe]:SPECtrum:FREQuency:CENTer

Sets or queries the center frequency in the spectrum measurement.

NOTE. *The center, start and stop frequencies are set interlocking each other with the following relationships: (start frequency) = (center frequency) - (span)/2 and (stop frequency) = (center frequency) + (span)/2.*

Conditions Measurement views: Spectrum

Group Sense commands

Syntax
```
[SENSe]:SPECtrum:FREQuency:CENTer <value>
[SENSe]:SPECtrum:FREQuency:CENTer?
```

Related Commands [SENSe]:SPECtrum:FREQuency:STARt, [SENSe]:SPECtrum:FREQuency:STOP

Arguments `<value> ::` `<NRf>` specifies the center frequency.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `SENSE:SPECTRUM:FREQUENCY:CENTER 7.5GHz` sets the center frequency to 7.5 GHz.

[SENSe]:SPECtrum:FREQuency:SPAN

Sets or queries the frequency span in the spectrum measurement.

Conditions Measurement views: Spectrum

Group Sense commands

Syntax
```
[SENSe]:SPECtrum:FREQuency:SPAN <value>
[SENSe]:SPECtrum:FREQuency:SPAN?
```

Arguments `<value> ::` `<NRf>` specifies the frequency span.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `SENSE:SPECTRUM:FREQUENCY:SPAN 20MHz` sets the span to 20 MHz.

[SENSe]:SPECtrum:FREQuency:SPAN:BANDwidth[:RESolution]:RATio

Sets or queries the ratio of span to RBW (Resolution Bandwidth) in the spectrum measurement. This command is valid when [SENSe]:SPECtrum:{BANDwidth|BWIDth}[:RESolution]:AUTO is set to On.

Conditions Measurement views: Spectrum

Group Sense commands

Syntax

```
[SENSe]:SPECtrum:FREQuency:SPAN:BANDwidth[:RESolution]:RATio
<value>
[SENSe]:SPECtrum:FREQuency:SPAN:BANDwidth[:RESolution]:
RATio?
```

Arguments `<value> ::` `<NRf>` specifies the ratio of span to RBW. Range: 20 to 1000.

Programming a specified ratio sets the RBW (= span/ratio), which is rounded down to the nearest valid value.

Examples `SENSE:SPECTRUM:FREQUENCY:SPAN:BANDWIDTH:RESOLUTION:RATIO 200` sets the ratio to 200, setting the RBW to 200 kHz for the span of 40 MHz.

[SENSe]:SPECtrum:FREQuency:STARt

Sets or queries the measurement start frequency (left edge on the graph) in the spectrum measurement.

The center, start and stop frequencies are set interlocking each other. Refer to the [SENSe]:SPECtrum:FREQuency:CENTer command.

Conditions Measurement views: Spectrum

Group Sense commands

Syntax
```
[SENSe]:SPECtrum:FREQuency:STARt <value>
[SENSe]:SPECtrum:FREQuency:STARt?
```

Related Commands [SENSe]:SPECtrum:FREQuency:STOP

Arguments `<value> ::` `<NRf>` is the measurement start frequency.
Range: (center frequency) ±(span)/2.

Examples `SENSE:SPECTRUM:FREQUENCY:START 6.95GHz` sets the start frequency to 6.95 GHz.

[SENSe]:SPECtrum:FREQuency:STEP

Sets or queries the frequency step size (the amount per press by which the up or down key changes the setting value). Programming a specified step size sets [SENSe]:SPECtrum:FREQuency:STEP:AUTO OFF.

Conditions Measurement views: Spectrum

Group Sense commands

Syntax
```
[SENSe]:SPECtrum:FREQuency:STEP <value>
[SENSe]:SPECtrum:FREQuency:STEP?
```

Related Commands [SENSe]:SPECtrum:FREQuency:STEP:AUTO

Arguments `<value> ::` `<NRf>` specifies the frequency step size.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `SENSE:SPECTRUM:FREQUENCY:STEP 1.5kHz` sets the step size to 1.5 kHz.

[SENSe]:SPECtrum:FREQuency:STEP:AUTO

Determines whether to set the frequency step size automatically or manually.

Conditions Measurement views: Spectrum

Group Sense commands

Syntax

```
[SENSe]:SPECtrum:FREQuency:STEP:AUTO { OFF | ON | 0 | 1 }
[SENSe]:SPECtrum:FREQuency:STEP:AUTO?
```

Arguments `OFF` or `0` specifies that the frequency step size is set manually using the [SENSe]:SPECtrum:FREQuency:STEP command.

`ON` or `1` specifies that the frequency step size is set automatically.

Examples `SENSE:SPECTRUM:FREQUENCY:STEP:AUTO ON` sets the frequency step size automatically.

[SENSe]:SPECtrum:FREQuency:STOP

Sets or queries the measurement stop frequency (right edge on the graph) in the spectrum measurement.

The center, start and stop frequencies are set interlocking each other. Refer to the [SENSe]:SPECtrum:FREQuency:CENTer command.

Conditions Measurement views: Spectrum

Group Sense commands

Syntax

```
[SENSe]:SPECtrum:FREQuency:STOP <value>
[SENSe]:SPECtrum:FREQuency:STOP?
```

Related Commands [SENSe]:SPECtrum:FREQuency:STARt

Arguments `<value> ::= <NRf>` is the measurement start frequency.
Range: (center frequency) ± (span)/2.

Examples `SENSE:SPECTRUM:FREQUENCY:STOP 7.05GHz` sets the stop frequency to 7.05 GHz.

[SENSe]:SPECtrum:LENGth

Sets or queries the spectrum length when [SENSe]:SPECtrum:TIMe:MODe is INDependent. Programming a specified length sets [SENSe]:SPECtrum:LENGth:AUTO OFF.

Conditions Measurement views: All

Group Sense commands

Syntax

```
[SENSe]:SPECtrum:LENGth <value>
[SENSe]:SPECtrum:LENGth?
```

Related Commands [SENSe]:SPECtrum:LENGth:AUTO, [SENSe]:SPECtrum:STARt

Arguments `<value> :: <NRf>` specifies the spectrum length.
Range: 0 to [(acquisition length) - (spectrum start)].

If [(spectrum start) + (spectrum length)] > (acquisition length), the actual spectrum length is reduced to the acquisition length.

Examples `SENSE:SPECTRUM:LENGTH 38.5us` sets the spectrum length to 38.5 µs.

[SENSe]:SPECtrum:LENGth:ACTual? (Query Only)

Queries the actual spectrum length.

Conditions Measurement views: All

Group Sense commands

Syntax

```
[SENSe]:SPECtrum:LENGth:ACTual?
```

Arguments None

Returns <NRf> Actual spectrum length in seconds.

Examples SENSE:SPECTRUM:LENGTH:ACTUAL? might return 6.337E-6, indicating that the actual spectrum length is 6.337 µs.

[SENSe]:SPECtrum:LENGth:AUTO

Sets or queries the spectrum length mode (automatic or manual) when [SENSe]:SPECtrum:TIMe:MODe is INDependent.

Conditions Measurement views: All

Group Sense commands

Syntax
```
[SENSe]:SPECtrum:LENGth:AUTO { OFF | ON | 0 | 1 }
[SENSe]:SPECtrum:LENGth:AUTO?
```

Arguments OFF or 0 sets the spectrum length manually, using the [SENSe]:SPECtrum:LENGth command.

ON or 1 sets the spectrum length automatically.

Examples SENSE:SPECTRUM:LENGTH:AUTO ON sets the spectrum length automatically.

[SENSe]:SPECtrum:MAX:SPAN (No Query Form)

Sets the frequency span to the maximum full span instead of the maximum real-time span. The maximum span depends on the model.

Conditions Measurement views: Spectrum

Group Sense commands

Syntax
```
[SENSe]:SPECtrum:MAX:SPAN
```

Arguments None

Examples `SENSE:SPECTRUM:MAX:SPAN` sets the frequency span to the maximum span.

[SENSe]:SPECtrum:POINts:COUNt

Sets or queries the number of sample points on the signal spectrum.

Conditions Measurement views: Spectrum

Group Sense commands

Syntax

```
[SENSe]:SPECtrum:POINts:COUNt { P801 | P1601 | P2401 | P3201
| P4001 | P6401 | P8001 | P10401 }
[SENSe]:SPECtrum:POINts:COUNt?
```

Arguments `P801` sets the number of sample points to 801.

`P1601` sets the number of sample points to 1601.

`P2401` sets the number of sample points to 2401.

`P3201` sets the number of sample points to 3201.

`P4001` sets the number of sample points to 4001.

`P6401` sets the number of sample points to 6401.

`P8001` sets the number of sample points to 8001.

`P10401` sets the number of sample points to 10401.

Examples `SENSE:SPECTRUM:POINTS:COUNT P801` sets the number of sample points to 801.

[SENSe]:SPECtrum:STARt

Sets or queries the spectrum offset time when [SENSe]:SPECtrum:TIMe:MODe is INDependent.

Conditions Measurement views: All

Group Sense commands

Syntax

```
[SENSe]:SPECtrum:STARt <value>
[SENSe]:SPECtrum:STARt?
```

Related Commands [SENSe]:SPECtrum:LENGth

Arguments `<value> :: <NRf>` specifies the spectrum offset time.
Range: 0 to the acquisition length.

If [(spectrum start) + (spectrum length)] > (acquisition length), the actual spectrum length is reduced to the acquisition length.

Examples `SENSE:SPECTRUM:START 23.5us` sets the analysis offset to 23.5 µs.

[SENSe]:SPECtrum:STARt:AUTO

Sets or queries the spectrum offset mode (automatic or manual) when [SENSe]:SPECtrum:TIMe:MODe is INDependent.

Conditions Measurement views: All

Group Sense commands

Syntax

```
[SENSe]:SPECtrum:STARt:AUTO { OFF | ON | 0 | 1 }
[SENSe]:SPECtrum:STARt:AUTO?
```

Arguments `OFF` or `0` sets the spectrum offset manually.

`ON` or `1` sets the spectrum length automatically.

Examples `SENSE:SPECTRUM:START:AUTO ON` sets the spectrum offset automatically.

[SENSe]:SPECtrum:TIMe:MODe

Sets or queries the spectrum time parameter mode (automatic or manual).

Conditions Measurement views: All

Group Sense commands

Syntax

```
[SENSe]:SPECtrum:TIMe:MODe { INDependent | COMMon }
[SENSe]:SPECtrum:TIMe:MODe?
```

Related Commands [SENSe]:ANALysis commands

Arguments `INDependent` sets the spectrum time parameters manually, using the [SENSe]:SPECtrum:STARt and [SENSe]:SPECtrum:LENGth commands.

`COMMon` sets the spectrum time parameters automatically using the analysis time settings.

Examples `SENSE:SPECTRUM:TIME:MODE COMMon` sets the spectrum time parameters automatically.

[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}

Sets or queries the channel bandwidth for the carrier as power reference. This command is valid when [SENSe]:SPURious:REFerence is set to CARRier.

Conditions Measurement views: Spurious

Group Sense commands

Syntax

```
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth} <value>
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}?
```

Arguments `<value> :: <NRf>` specifies the channel bandwidth.
Range: 100 Hz to 40 MHz (Standard) / 110 MHz (Option 110).

Examples `SENSE:SPURIOUS:CARRIER:BANDWIDTH 20MHz` sets the channel bandwidth to 20 MHz.

[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}:INTegration

Sets or queries the integration bandwidth to calculate the carrier power. This command is valid when [SENSe]:SPURious:REFerence is set to CARRier.

Conditions Measurement views: Spurious

Group Sense commands

Syntax
```
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}:INTegration
<value>
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}:INTegration?
```

Arguments <value> :: <NRf> specifies the integration bandwidth.
Range: 100 Hz to 40 MHz (Standard) / 110 MHz (Option 110).

Examples SENSE:SPURIOUS:CARRIER:BANDWIDTH:INTEGRATION 2MHz sets the integration bandwidth to 2 MHz.

[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}[:RESolution]

Sets or queries the resolution bandwidth (RBW) to measure the carrier power. This command is valid when [SENSe]:SPURious:REFerence is set to CARRier. Programming a specified RBW sets [SENSe]:SPURious:CARRier:BANDwidth [RESolution]:AUTO OFF.

Conditions Measurement views: Spurious

Group Sense commands

Syntax
```
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}[:RESolution]
<value>
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}[:RESolution]?
```

Related Commands [SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}[:RESolution]:AUTO

Arguments <value> :: <NRf> specifies the RBW. Range: 1 Hz to 10 MHz.

Examples SENSE:SPURIOUS:CARRIER:BANDWIDTH:RESOLUTION 200kHz sets the RBW to 200 kHz.

[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}[:RESolution]:AUTO

Determines whether to set the resolution bandwidth (RBW) automatically or manually when the power reference is set to carrier ([SENSe]:SPURious:REFerence is set to CARRier) in the Spurious measurement.

Conditions Measurement views: Spurious

Group Sense commands

Syntax
```
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}[:RESolution]:
AUTO { OFF | ON | 0 | 1 }
[SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}[:RESolution]:
AUTO?
```

Arguments `OFF` or `0` specifies that the resolution bandwidth is set manually using the [SENSe]:SPURious:CARRier:{BANDwidth|BWIDth}[:RESolution] command.

`ON` or `1` specifies that the resolution bandwidth is set automatically.

Examples `SENSE:SPURIOUS:CARRIER:BANDWIDTH:RESOLUTION:AUTO ON` sets the resolution bandwidth automatically.

[SENSe]:SPURious:CARRier:DETection

Sets or queries the carrier detection method. This command is valid when [SENSe]:SPURious:REFerence is set to CARRier.

Conditions Measurement views: Spurious

Group Sense commands

Syntax
```
[SENSe]:SPURious:CARRier:DETection { AVERage | PEAK | AVGLog
}
[SENSe]:SPURious:CARRier:DETection?
```

Arguments `AVERage` selects the average detection.

`PEAK` selects the peak detection.

`AVGLog` selects the average (of logs) detection.

Examples `SENSE:SPURIOUS:CARRIER:DETECTION PEAK` selects the peak detection.

[SENSe]:SPURious:CARRier:FREQuency

Sets or queries the carrier frequency in the Spurious measurement. This command is valid when [SENSe]:SPURious:REFerence is set to CARRier.

Conditions Measurement views: Spurious

Group Sense commands

Syntax

```
[SENSe]:SPURious:CARRier:FREQuency <value>
[SENSe]:SPURious:CARRier:FREQuency?
```

Arguments `<value> ::` `<NRf>` specifies the carrier frequency.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `SENSE:SPURIOUS:CARRIER:FREQUENCY 7.5GHz` sets the carrier frequency to 7.5 GHz.

[SENSe]:SPURious:CARRier:THReshold

Sets or queries the threshold level to detect the carrier in the Spurious measurement. This command is valid when [SENSe]:SPURious:REFerence is set to CARRier.

Conditions Measurement views: Spurious

Group Sense commands

Syntax

```
[SENSe]:SPURious:CARRier:THReshold <value>
[SENSe]:SPURious:CARRier:THReshold?
```

Arguments `<value> ::` `<NRf>` specifies the threshold level above which the input signal is determined to be a carrier. Range: -170 to +50 dBm. The unit can be changed by the [SENSe]:POWer:UNITs or UNIT:POWer command.

Examples `SENSE:SPURIOUS:CARRIER:THRESHOLD -25` sets the carrier threshold level to -25 dBm.

[SENSe]:SPURious:CLEar:RESults (No Query Form)

Restarts multi-trace functions (Average and Max Hold).

Conditions Measurement views: Spurious

Group Sense commands

Syntax `[SENSe]:SPURious:CLEar:RESults`

Arguments None

Examples `SENSE:SPURIOUS:CLEAR:RESULTS` restarts multi-trace functions.

[SENSe]:SPURious[:FREQuency]:OVERlap? (Query Only)

Queries whether any of the frequency ranges (A to T) overlap, including the carrier when [SENSe]:SPURious:REFerence is set to CARRier.

NOTE. *If there are any overlaps between the ranges, the measurement will not run correctly.*

Conditions Measurement views: Spurious

Group Sense commands

Syntax `[SENSe]:SPURious[:FREQuency]:OVERlap?`

Arguments None

Returns `0` (no overlap) or `1` (overlap).

Examples `SENSE:SPURIOUS:FREQUENCY:OVERLAP?` might return `1`, indicating that some frequency ranges overlap.

[SENSe]:SPURious:LIST

Sets or queries how to list the spurious signals in the Spurious measurement.

Conditions Measurement views: Spurious

Group Sense commands

Syntax

```
[SENSe]:SPURious:LIST { ALL | OVERlimit }
[SENSe]:SPURious:LIST?
```

Arguments `ALL` lists all of the detected spurious signals.

`OVERlimit` lists the spurious signals exceeding the limits. Use the [SENSe]:SPURious:RANGe<x>:LIMit command group to set the limits.

Examples `SENSE:SPURIOUS:LIST ALL` lists all of the detected spurious signals.

[SENSe]:SPURious:MODE

Sets or queries the frequency range mode in the Spurious measurement.

Conditions Measurement views: Spurious

Group Sense commands

Syntax

```
[SENSe]:SPURious:MODE { MULTi | SINGle }
[SENSe]:SPURious:MODE?
```

Arguments `MULTi` displays all of the ranges that are enabled. Use the [SENSe]:SPURious:RANGe<x>:STATe command to enable the range.

`SINGle` displays only the range that the selected spurious signal is in. Use the DISPlay:SPURious:SELect:NUMBer command to select the spurious signal.

Examples `SENSE:SPURIOUS:MODE MULTi` displays all of the enabled ranges.

[SENSe]:SPURious:OPTimization

Sets or queries the method of optimizing the gain and input bandwidth in the Spurious measurement.

Conditions Measurement views: Spurious

Group Sense commands

Syntax

```
[SENSe]:SPURious:OPTimization { AUTO | MINTime | MAXDynrange
| MINNoise }
[SENSe]:SPURious:OPTimization?
```

Arguments

`AUTO` optimizes automatically the gain and input bandwidth.

`MINTime` optimizes the gain and input bandwidth to minimize sweep time.

`MAXDynrange` optimizes the gain and input bandwidth to maximize the dynamic range.

`MINNoise` optimizes the gain and input bandwidth to minimize noise.

Examples `SENSE:SPURIOUS:OPTIMIZATION AUTO` optimizes automatically the gain and input bandwidth.

[SENSe]:SPURious:POINts:COUNt

Sets or queries the number of sample points on the spectrum trace per range in the Spurious measurement.

Conditions Measurement views: Spurious

Group Sense commands

Syntax

```
[SENSe]:SPURious:POINts:COUNt { P801 | P2401 | P4001 | P8001
| P10401 }
[SENSe]:SPURious:POINts:COUNt?
```

Arguments

`P801` sets the trace points to 801 per range.

`P2401` sets the trace points to 2401 per range.

`P4001` sets the trace points to 4001 per range.

P8001 sets the trace points to 8001 per range.

P10401 sets the trace points to 10401 per range.

Examples SENSE:SPURIOUS:POINTS:COUNT P801 sets the trace points to 801 per range.

[SENSe]:SPURious:RANGe<x>:BANDwidth:VIDeo

Sets or queries the video bandwidth (VBW) in the specified frequency range. Programming a specified VBW sets [SENSe]:SPURious:BANDwidth:VIDeo STATe OFF.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions Measurement views: Spurious

Group Sense commands

Syntax
```
[SENSe]:SPURious:RANGe<x>:BANDwidth:VIDeo <value>
[SENSe]:SPURious:RANGe<x>:BANDwidth:VIDeo?
```

Related Commands [SENSe]:SPECtrum:{BANDwidth|BWIDth}:VIDeo:STATe

Arguments <value> :: <NRf> specifies the VBW.
Range: Current RBW/10^4 (1 Hz minimum) to Current RBW.

Examples SENSE:SPURIOUS:RANGE1:BANDWIDTH:VIDEO 200kHz sets the VBW to 200 kHz for Range A.

[SENSe]:SPURious:RANGe<x>:BANDwidth:VIDeo:STATe

Determines whether to enable or disable the video bandwidth (VBW) in the specified frequency range.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions Measurement views: Spurious

Group Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:BANDwidth:VIDeo:STATe { OFF | ON
| 0 | 1 }
[SENSe]:SPURious:RANGe<x>:BANDwidth:VIDeo:STATe?
```

Arguments

`OFF` or `0` disables the VBW.

`ON` or `1` enables the VBW.

Examples

`SENSE:SPURIOUS:RANGE1:BANDWIDTH:VIDEO:STATE ON` enables the VBW for Range A.

[SENSe]:SPURious:RANGe<x>:DETection

Sets or queries the spurious detection method in the specified frequency range.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:DETection { AVERage | PEAK |
QUASipeak | CAVerage | CPEak | AVGLog }
[SENSe]:SPURious:RANGe<x>:DETection?
```

Arguments

`AVERage` selects the average detection.

`PEAK` selects the peak detection.

`QUASipeak` selects the quasi-peak detection.

`CAVerage` selects the CISPR average detection.

`CPEak` selects the CISPR peak detection.

`AVGLog` selects the average (of logs) detection.

Examples

`SENSE:SPURIOUS:RANGE1:DETECTION PEAK` selects the peak detection for Range A.

[SENSe]:SPURious:RANGe<x>:EXCursion

Sets or queries the excursion level (how far down the signal must drop between spurious emissions) in the specified frequency range.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions Measurement views: Spurious

Group Sense commands

Syntax
```
[SENSe]:SPURious:RANGe<x>:EXCursion <value>
[SENSe]:SPURious:RANGe<x>:EXCursion?
```

Arguments `<value> ::= <NRf>` specifies the excursion level. A signal with amplitude less than the excursion level is considered to be a noise. Range: 1 to 50 dB.

Examples `SENSE:SPURIOUS:RANGE1:EXCURSION 8` sets the excursion level to 8 dB.

[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]

Sets or queries the filter shape to search the specified frequency range for spurious signals.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions Measurement views: Spurious

Group Sense commands

Syntax
```
[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe] { RBW | MIL6db |
CISPr }
[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]?
```

Arguments `RBW` selects the RBW filter.

`MIL6db` selects the -6 dB RBW (MIL) filter.

`CISPr` selects the CISPR filter.

Examples `SENSE:SPURIOUS:RANGE1:FILTER:SHAPE CISPr` uses the CISPR filter in Range A.

[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]:BANDwidth

Sets or queries the filter bandwidth to search the specified frequency range for spurious signals.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions Measurement views: Spurious

Group Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]:BANDwidth <value>
[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]:BANDwidth?
```

Arguments `<value> ::  <NRf>` specifies the filter bandwidth.

The setting range varies depending on the center frequency and trace detector when using the CISPR filters and detectors. Otherwise, the bandwidth minimum is a function of the span (= (stop frequency) - (start frequency)) of the range and the maximum is fixed at 5 MHz.

By default, the value is set automatically ([SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]:BANDwidth:AUTO is ON). When the detection is set to CISPR QPk ([SENSe]:SPURious:RANGe<x>:DETection is QUASipeak), only one value is allowed, which is set automatically.

Examples `SENSE:SPURIOUS:RANGE1:FILTER:SHAPE:BANDWIDTH 200kHz` sets the filter bandwidth to 200 kHz for Range A.

[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]:BANDwidth:AUTO

Determines whether to set the filter bandwidth automatically or manually for the specified frequency range in the Spurious measurement.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions Measurement views: Spurious

Group Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]:BANDwidth:AUTO {
OFF | ON | 0 | 1 }
[SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]:BANDwidth:AUTO?
```

Arguments

`OFF` or `0` specifies that the filter bandwidth is set manually using the [SENSe]:SPURious:RANGe<x>:FILTer[:SHAPe]:BANDwidth command.

`ON` or `1` specifies that the filter bandwidth is set automatically.

Examples

`SENSE:SPURIOUS:RANGE1:FILTER:SHAPE:BANDWIDTH:AUTO ON` sets the filter bandwidth automatically for Range A.

[SENSe]:SPURious:RANGe<x>:FREQuency:STARt

Sets or queries the start frequency of the specified range in the Spurious measurement.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

NOTE. *The frequency ranges must not be overlapped. Use the [SENSe]:SPURious[:FREQuency]:OVERlap? query to check whether there is any overlap.*

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:FREQuency:STARt <value>
[SENSe]:SPURious:RANGe<x>:FREQuency:STARt?
```

Arguments

`<value> :: <NRf>` specifies the start frequency.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples

`SENSE:SPURIOUS:RANGE1:FREQUENCY:START 1.615GHz` sets the start frequency of Range A to 1.615 GHz.

[SENSe]:SPURious:RANGe<x>:FREQuency:STOP

Sets or queries the stop frequency of the specified range in the Spurious measurement.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

NOTE. *The frequency ranges must not be overlapped. Use the [SENSe]:SPURious[:FREQuency]:OVERlap? query to check whether there is any overlap.*

Conditions Measurement views: Spurious

Group Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:FREQuency:STOP <value>
[SENSe]:SPURious:RANGe<x>:FREQuency:STOP?
```

Arguments `<value> ::` `<NRf>` specifies the stop frequency.
Range:

- RSA5103B – 0 to 3 GHz
- RSA5106B – 0 to 6.2 GHz
- RSA5115B – 0 to 15 GHz
- RSA5126B – 0 to 26.5 GHz

Examples `SENSE:SPURIOUS:RANGE1:FREQUENCY:STOP 1.715GHz` sets the stop frequency of Range A to 1.715 GHz.

[SENSe]:SPURious:RANGe<x>:LIMit:ABSolute:STARt

Sets or queries the absolute amplitude of the limits at the start (left edge) of the specified range in the Spurious measurement. This command is valid when [SENSe]:SPURious:RANGe<x>:LIMit:MASK is set to ABS, AND, or OR.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions Measurement views: Spurious

Group Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:LIMit:ABSolute:STARt <value>
[SENSe]:SPURious:RANGe<x>:LIMit:ABSolute:STARt?
```

Arguments

<value> :: <NRf> specifies the absolute start amplitude of the limits. Range: -170 to +50 dBm.
The unit can be changed by the [SENSe]:POWer:UNITs or UNIT:POWer command.

Examples

SENSE:SPURIOUS:RANGE1:LIMIT:ABSOLUTE:START -30 sets the absolute start amplitude of the limits for Range A to -30 dBm.

[SENSe]:SPURious:RANGe<x>:LIMit:ABSolute:STOP

Sets or queries the absolute amplitude of the limits at the stop (right edge) of the specified range in the Spurious measurement. This command is valid when [SENSe]:SPURious:RANGe<x>:LIMit:MASK is set to ABS, AND, or OR.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions

Measurement views: Spurious

Group

Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:LIMit:ABSolute:STOP <value>
[SENSe]:SPURious:RANGe<x>:LIMit:ABSolute:STOP?
```

Related Commands

[SENSe]:POWer:UNITs, UNIT:POWer

Arguments

<value> :: <NRf> specifies the absolute stop amplitude of the limits. Range: -170 to +50 dBm.
The unit can be changed by the [SENSe]:POWer:UNITs or UNIT:POWer command.

Examples

SENSE:SPURIOUS:RANGE1:LIMIT:ABSOLUTE:STOP -10 sets the absolute stop amplitude of the limits for Range A to -10 dBm.

[SENSe]:SPURious:RANGe<x>:LIMit:MASK

Sets or queries the limit mask function mode for the specified range in the Spurious measurement.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions Measurement views: Spurious

Group Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:LIMit:MASK { ABS | REL | AND | OR
| OFF }
[SENSe]:SPURious:RANGe<x>:LIMit:MASK?
```

Arguments The following table lists the arguments.

Table 2-37: Limit mask mode

Argument	Description
ABS	Failure is detected when one of the spurious signals is larger than the absolute amplitude limit.
REL	Failure is detected when one of the spurious signals is larger than the relative amplitude limit.
AND	Failure is detected when one of the spurious signals is larger than the absolute AND relative amplitude limits.
OR	Failure is detected when one of the spurious signals is larger than the absolute OR relative amplitude limit.
OFF	Disables the mask.

To set the absolute amplitude limits, use the [SENSe]:SPURious:RANGe<x>:LIMit:ABSolute:STARt and [SENSe]:SPURious:RANGe<x>:LIMit:ABSolute:STOP commands.

To set the relative amplitude limits, use the [SENSe]:SPURious:RANGe<x>:LIMit:RELative:STARt and [SENSe]:SPURious:RANGe<x>:LIMit:RELative:STOP commands.

Examples `SENSE:SPURIOUS:RANGE1:LIMIT:MASK ABS` specifies that failure is detected when one of the spurious signals is larger than the absolute amplitude limit in Range A.

[SENSe]:SPURious:RANGe<x>:LIMit:RELative:STARt

Sets or queries the relative amplitude of the limits at the start (left edge) of the specified range in the Spurious measurement. This command is valid when [SENSe]:SPURious:RANGe<x>:LIMit:MASK is set to REL, AND, or OR.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions Measurement views: Spurious

Group Sense commands

Syntax
```
[SENSe]:SPURious:RANGe<x>:LIMit:RELative:STARt <value>
[SENSe]:SPURious:RANGe<x>:LIMit:RELative:STARt?
```

Arguments <value> :: <NRf> specifies the relative start amplitude of the limits. Range: -100 to 0 dB.

Use the [SENSe]:SPURious:REFerence command to select the power reference.

Examples SENSE:SPURIOUS:RANGE1:LIMIT:RELATIVE:START -30 sets the relative start amplitude of the limits for Range A to -30 dB.

[SENSe]:SPURious:RANGe<x>:LIMit:RELative:STOP

Sets or queries the relative amplitude of the limits at the stop (right edge) of the specified range in the Spurious measurement. This command is valid when [SENSe]:SPURious:RANGe<x>:LIMit:MASK is set to REL, AND, or OR.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions Measurement views: Spurious

Group Sense commands

Syntax
```
[SENSe]:SPURious:RANGe<x>:LIMit:RELative:STOP <value>
[SENSe]:SPURious:RANGe<x>:LIMit:RELative:STOP?
```

Arguments <value> :: <NRf> specifies the relative stop amplitude of the limits. Range: -100 to 0 dB.

Use the [SENSe]:SPURious:REFerence command to select the power reference.

Examples SENSE:SPURIOUS:RANGE1:LIMIT:RELATIVE:STOP -10 sets the relative stop amplitude of the limits for Range A to -10 dB.

[SENSe]:SPURious:RANGe<x>:STATe

Determines whether to enable or disable the frequency range in the Spurious measurement.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions Measurement views: Spurious

Group Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:STATe { OFF | ON | 0 | 1 }
[SENSe]:SPURious:RANGe<x>:STATe?
```

Arguments `OFF` or `0` disables the frequency range.

`ON` or `1` enables the frequency range.

Examples `SENSE:SPURIOUS:RANGE1:STATE ON` enables Range A.

[SENSe]:SPURious:RANGe<x>:THReshold

Sets or queries the threshold level to detect spurious signals in the specified frequency range.

The parameter <x> = 1 to 20, representing Range A to T, respectively.

Conditions Measurement views: Spurious

Group Sense commands

Syntax

```
[SENSe]:SPURious:RANGe<x>:THReshold <value>
[SENSe]:SPURious:RANGe<x>:THReshold?
```

Arguments `<value> ::= <NRf>` specifies the threshold level above which the signal is determined to be spurious. Range: -50 to +30 dBm.

The unit can be changed by the [SENSe]:POWer:UNITs or UNIT:POWer command.

Examples `SENSE:SPURIOUS:RANGE1:THRESHOLD -25` sets the threshold level to -25 dBm in Range A.

[SENSe]:SPURious:REFerence

Sets or queries the power reference in the Spurious measurement.

Conditions Measurement views: Spurious

Group Sense commands

Syntax

```
[SENSe]:SPURious:REFerence { CARRier | MANual | NONE }
[SENSe]:SPURious:REFerence?
```

Arguments `CARRier` uses the carrier as the power reference.

`MANual` sets the power reference using the [SENSe]:SPURious:REFerence:MANual:POWer command.

`NONE` uses no reference.

Examples `SENSE:SPURIOUS:REFERENCE CARRier` uses the carrier as the power reference.

[SENSe]:SPURious:REFerence:MANual:POWer

Sets or queries the reference power level in the Spurious measurement. This command is valid when [SENSe]:SPURious:REFerence is set to MANual.

Conditions Measurement views: Spurious

Group Sense commands

Syntax

```
[SENSe]:SPURious:REFerence:MANual:POWer <value>
[SENSe]:SPURious:REFerence:MANual:POWer?
```

Arguments `<value> ::= <NRf>` specifies the reference power level. Range: -170 to +50 dBm. The unit can be changed by the [SENSe]:POWer:UNITs or UNIT:POWer command.

Examples `SENSE:SPURIOUS:REFERENCE:MANUAL:POWER -25` sets the reference power level to -25 dBm.

[SENSe]:SQUality:EVM:ALL[:STATe]

Sets or queries whether or not the display of both EVM and Offset EVM results is enabled. Offset EVM appears only for OQPSK and SOQPSK modulation types.

Conditions Measurement views: Signal Quality

Group Sense commands

Syntax

```
[SENSe]:SQUality:EVM:ALL[:STATe]
[SENSe]:SQUality:EVM:ALL[:STATe]?
```

Arguments OFF or 0 disables the local lockout operation.

ON or 1 enables the local lockout operation.

SENSe:TOFF:POWEr:SPECtral:DENSity

Sets or queries the limit set for comparison of the Toff power spectral density value in the Limits tab of the LTE Power vs Time display settings control panel.

Conditions Measurement view: LTE Power vs Time

Group Sense commands

Syntax

```
SENSe:TOFF:POWEr:SPECtral:DENSity <value>
SENSe:TOFF:POWEr:SPECtral:DENSity?
```

Arguments <value>::<NRf> specifies the Toff power spectral density value.

Returns <NRf>

Examples `SENSE:TOFF:POWER:SPECTRAL:DENSITY` might return –83.000000000, indicating that the Toff power spectral density is set to –83 dBm.

[SENSe]:TOVerview:FREQuency:CENTer

Sets or queries the center frequency in the time overview.

Conditions Measurement views: Time overview

Group Sense commands

Syntax
```
[SENSe]:TOVerview:FREQuency:CENTer <value>
[SENSe]:TOVerview:FREQuency:CENTer?
```

Arguments `<value> ::  <NRf>` specifies the center frequency.
Range:

- RSA5103B – 1 to 3 GHz
- RSA5106B – 1 to 6.2 GHz
- RSA5115B – 1 to 15 GHz
- RSA5126B – 1 to 26.5 GHz

Examples `SENSE:TOVERVIEW:FREQUENCY:CENTER 7.5GHz` sets the center frequency to 7.5 GHz.

[SENSe]:TOVerview:MAXTracepoints

Sets or queries the maximum trace points in the time overview.

Conditions Measurement views: Time overview

Group Sense commands

Syntax
```
[SENSe]:TOVerview:MAXTracepoints { ONEK | TENK | HUNDredk
| NEVerdecimate }
[SENSe]:TOVerview:MAXTracepoints?
```

Arguments `ONEK` sets the maximum trace points to 1 k.

`TENK` sets the maximum trace points to 10 k.

`HUNDredk` sets the maximum trace points to 100 k.

NEVerdecimate never decimates the trace points.

Examples SENSE:TOVERVIEW:MAXTRACEPOINTS TENK sets the maximum trace points to 10 k.

[SENSe]:TOVerview:TIME:PER:DIVision

This command sets or queries the time per division (dB/div) value in the Time Overview display.

Conditions Measurement view: Time Overview

Group Sense commands

Syntax
```
[SENSe]:TOVerview:TIME:PER:DIVision <NRf>
[SENSe]:TOVerview:TIME:PER:DIVision?
```

Arguments <NRf> is the dB/div value.

Range: .01 to 20

Returns <NR2> = floating point value without an exponent, which is the time per division value.

Examples SENSE:TOV:TIME:PER:DIV? might return 10.0000000000, indicating that the time per division value is set to 10.0 dB.

SENSe:TXGain:{BANDwidth|BWIDth}[:RESolution]

Sets or queries the resolution bandwidth.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax
```
SENSe:TXGain:{BANDwidth|BWIDth}[:RESolution] <value>
SENSe:TXGain:{BANDwidth|BWIDth}[:RESolution]?
```

Arguments <value>::=<NRf> specifies the RBW.

Range: 1 kHz to 1 MHz.

Returns See Arguments.

Examples `SENSe:TXGain:BANDwidth:RESolution 200E3` sets the RBW to 200 kHz.

SENSe:TXGain:FREQuency:CENTer

Sets or queries the frequency at the center of the selected Span.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax
```
SENSe:TXGain:FREQuency:CENTer <value>
SENSe:TXGain:FREQuency:CENTer?
```

Arguments <value>::=<NRf> specifies the center frequency.

Range: 1 kHz to 1 MHz.

Returns See Arguments.

Examples `SENSE:TXGAIN:FREQUENCY:CENTER 1E9` sets the center frequency to 1 GHz.

SENSe:TXGain:FREQuency:POINts

Sets or queries the number of frequency steps in the measurement range.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax
```
SENSe:TXGain:FREQuency:POINts <value>
SENSe:TXGain:FREQuency:POINts?
```

Arguments <value>::=<NR1> specifies number of frequency steps.

Returns See Arguments.

Examples `SENSE:TXGAIN:FREQUENCY:POINTS 101` sets the number of frequency steps to 101.

SENSe:TXGain:FREQuency:SPAN

Sets or queries the difference between the start and stop frequencies (span).

Conditions Measurement view: Transmission Gain

Group Display group

Syntax
```
SENSe:TXGain:FREQuency:SPAN <value>
SENSe:TXGain:FREQuency:SPAN?
```

Arguments <value>::=<NR1> specifies the frequency span.

Returns See Arguments.

Examples `SENSE:TXGAIN:FREQUENCY:SPAN 100E6` sets the frequency span to 100 MHz.

SENSe:TXGain:FREQuency:STARt

Sets or queries the lowest frequency in the span.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax
```
SENSe:TXGain:FREQuency:STARt <value>
SENSe:TXGain:FREQuency:STARt?
```

Related Commands SENSe:TXGain:FREQuency:STOP

Arguments <value>::=<NR1> specifies the lowest frequency in the span.

Returns See Arguments.

Examples `SENSE:TXGAIN:FREQUENCY:START 1E9` sets the lowest frequency to 1 GHz.

SENSe:TXGain:FREQuency:STEP

Sets or queries the frequency increment. Measurements are made at frequency points from Start Freq to Stop Freq in Step Size increments.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax
```
SENSe:TXGain:FREQuency:STEP <value>
SENSe:TXGain:FREQuency:STEP?
```

Arguments <value>::=<NRf> specifies the frequency increment.

Returns See Arguments.

Examples `SENSE:TXGAIN:FREQUENCY:STEP 1E6` sets the frequency increment to 1 MHz.

SENSe:TXGain:FREQuency:STOP

Sets or queries the highest frequency in the span.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax
```
SENSe:TXGain:FREQuency:STOP <value>
SENSe:TXGain:FREQuency:STOP?
```

Related Commands SENSe:TXGain:FREQuency:STARt

Arguments <value>::=<NRf> specifies the highest frequency.

Returns See Arguments.

Examples `SENSE:TXGAIN:FREQUENCY:STOP 2E9` sets the highest frequency to 2 GHz.

[SENSe]:USETtings (No Query Form)

Updates the analyzer settings. This command is useful when you need to set the analyzer including the RF attenuation before taking data acquisition. Unless this command is executed, the attenuation value is not set until acquisition is taken.

Conditions Measurement views: All

Group Sense commands

Syntax `[SENSe]:USETtings`

Arguments None

Examples `SENSE:USETTINGS` updates settings.

[SENSe]:WLAN:ANALysis:LENGth

Specifies the length of the analysis period to use in the WLAN analysis time settings. Analysis length is specified in either symbols or seconds, depending on the Units setting. Use [SENSe]:WLAN:UNIT:TIME to set the units.

Programming a specified length sets `[SENSe]:WLAN:ANALysis:AUTO` to `OFF`.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax
```
[SENSe]:WLAN:ANALysis:LENGth <NRf>
[SENSe]:WLAN:ANALysis:LENGth?
```

Arguments Floating point number that specifies the analysis length.

Range: the minimum value depends on the modulation type.

Examples `SENSE:WLAN:ANALYSIS:LENGTH 25.625e-6` sets the analysis length to 25.625 µs.

[SENSe]:WLAN:ANALysis:LENGth:ACTual? (Query Only)

Returns the value of the actual analysis length for the WLAN analysis time settings.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax `[SENSe]:WLAN:ANALysis:LENGth:ACTual?`

Returns Floating point number that represents the actual analysis length in seconds.

Examples `SENSE:WLAN:ANALYSIS:LENGTH:ACTUAL?` might return 25.625E-6, indicating that the actual analysis length is 25.625 µs.

[SENSe]:WLAN:ANALysis:LENGth:AUTO

Sets the analysis length value based on the requirements of the selected display. To set the analysis length manually, use the command.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax
```
[SENSe]:WLAN:ANALysis:LENGth:AUTO {0|1|OFF|ON}
[SENSe]:WLAN:ANALysis:LENGth:AUTO?
```

Arguments `ON` or `1` sets the analysis length value based on the requirements of the selected display.

`OFF` or `0` turns this function off.

Examples `SENSE:WLAN:ANALYSIS:LENGTH:AUTO ON` sets the analysis length value automatically based on the requirements of the selected display.

[SENSe]:WLAN:ANALysis:OFFSet

Specifies or queries the analysis offset in the symbol interval in WLAN analysis parameters. This value is a percentage of the Guard Interval. The allowable range is -100% to 0%. -100% positions the FFT to start at the beginning of the Guard Interval, 0% positions it to start at the end of the Guard Interval. The default value is -50% which usually gives the best measurement results.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax
```
[SENSe]:WLAN:ANALysis:OFFSet <NR1>
[SENSe]:WLAN:ANALysis:OFFSet?
```

Arguments Integer that represents the analysis offset in the symbol interval, in percent terms.

Examples `SENSE:WLAN:ANALYSIS:OFFSET -50` sets the analysis offset in the symbol interval to -50%.

[SENSe]:WLAN:BURSt

Specifies the burst to measure when multiple bursts are present in an acquisition, in the WLAN analysis parameters. The index number of the first burst is 1, second burst is 2, etc.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax
```
[SENSe]:WLAN:BURSt <NR1>
[SENSe]:WLAN:BURSt?
```

Arguments Integer that specifies which burst to measure when multiple bursts are present in an acquisition.

Examples `SENSE:WLAN:BURST 3` sets the burst to measure in the presence of multiple bursts to the third burst.

[SENSe]:WLAN:CHANnel[:BANDwidth|:BWIDth]

Specifies or queries a nominal channel bandwidth to use for the WLAN measurements.

To automatically set the subcarrier spacing for the WLAN measurements, use the command [SENSe]:WLAN:SCARriers:SPACing:AUTO.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax
```
[SENSe]:WLAN:CHANnel[:BANDwidth|:BWIDth]
{BW5|BW10|BW20|BW40|BW80|BW160}
[SENSe]:WLAN:CHANnel[:BANDwidth|:BWIDth]?
```

Arguments `BW5, BW10, BW20, BW40, BW80, BW160` specify a nominal channel bandwidth to use for the WLAN measurements.

Returns Query returns one of the argument values indicating the selected nominal channel BW.

Examples `SENSE:WLAN:CHANNEL:BANDWIDTH BW20` sets the nominal channel bandwidth to use to 20 MHz.

[SENSe]:WLAN:EQUalizer:TRAining

Specifies the method to use for estimating channel frequency response and equalization for WLAN measurments (either PREamble or PDATa). This control can be used to diagnose changes in frequency response over the signal packet.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax

```
[SENSe]:WLAN:EQUalizer:TRAining {PREamble|PDATa}
[SENSe]:WLAN:EQUalizer:TRAining?
```

Arguments `PREamble` The instrument uses only the Preamble to estimate channel frequency response. This response is then used to equalize the entire signal packet.

`PDATa` Preamble plus data. The instrument makes an initial channel frequency response estimate from the Preamble. Then it estimates the channel response for each data symbol using the decoded data content to derive equalization for each symbol individually. This allows compensation for time-varying channel response over the packet.

Examples `SENSE:WLAN:EQUALIZER:TRAINING PREAMBLE` sets the method used to estimate channel frequency response and equalization to Preamble.

[SENSe]:WLAN:FFT:LENGth? (Query Only)

Returns the fixed FFT length for the WLAN measurements.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax

```
[SENSe]:WLAN:FFT:LENGth?
```

Returns Integer that represents the fixed FFT length for the WLAN measurements.

Examples `SENSE:WLAN:FFT:LENGTH?` might return the fixed FFT length as 64.

[SENSe]:WLAN:FFT:SRATe? (Query Only)

Returns the FFT sample rate for the WLAN measurements, in samples per second, based on the bandwidth or subcarrier setting.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax `[SENSe]:WLAN:FFT:SRATe?`

Returns Floating point number that represents the FFT sample rate in MS/s.

Examples `SENSE:WLAN:FFT:SRATE?` might return `20.000e+6`, which represents an FFT sample rate of 20.000 MS/s.

[SENSe]:WLAN:FREQuency:ERRor

Sets or queries the frequency error in the WLAN analysis parameters. The entered value is used by the analysis as a fixed frequency offset. This is useful when the exact frequency offset of the signal is known. To set the frequency error automatically, use the command [SENSe]:WLAN:FREQuency:ERRor:AUTO.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax
```
[SENSe]:WLAN:FREQuency:ERRor <NRf>
[SENSe]:WLAN:FREQuency:ERRor?
```

Arguments Floating point number that represents the fixed frequency offset.

Examples `[SENSE]:WLAN:FREQUENCY:ERROR 0.02` sets the frequency error to 0.02.

[SENSe]:WLAN:FREQuency:ERRor:AUTO

Turns on the automatic determination of the frequency error in the WLAN analysis parameters. To enter a value manually when Auto is turned off, use the command `[SENSe]:WLAN:FREQuency:ERRor`.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax
```
[SENSe]:WLAN:FREQuency:ERRor:AUTO {0|1|OFF|ON}
[SENSe]:WLAN:FREQuency:ERRor:AUTO?
```

Arguments `OFF` or `0` sets the frequency error manually, using the `[SENSe]:WLAN:FREQuency:ERRor` command.

`ON` or `1` sets the frequency error automatically.

Examples `SENSE:WLAN:FREQUENCY:ERROR:AUTO ON` Turns on automatic determination of the frequency error.

[SENSe]:WLAN:GUARd:INTerval

Sets or queries the Guard Interval to use in the input signal in the WLAN measurements.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax
```
[SENSe]:WLAN:GUARd:INTerval {AUTO|FOURth|EIGHth}
[SENSe]:WLAN:GUARd:INTerval?
```

Arguments AUTO Auto from SIG uses the Guard Interval value extracted from the signal

FOURth specifies the guard interval as 1/4.

EIGHth specifies the guard interval as 1/8.

Examples SENSE:WLAN:GUARD:INTERVAL FOUR sets the guard interval to 1/4.

[SENSe]:WLAN:MODulation:MANual

Specifies to use a fixed modulation type, regardless of the signal content, in the WLAN analysis parameters.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax
```
[SENSe]:WLAN:MODulation:MANual
{BPSK|QPSK|QAM16|QAM64|QAM256|DSSS1|DSSS2|CCK5|CCK11}
[SENSe]:WLAN:MODulation:MANual?
```

Arguments BPSK|QPSK|QAM16|QAM64|QAM256|DSSS1|DSSS2|CCK5|CCK11 are fixed modulation types.

Examples [SENSe]:WLAN:MODulation:MANual DSSS1 specifies to use DSSS1 fixed modulation regardless of the signal content.

[SENSe]:WLAN:MODulation:SOURce

Specifies the method of data symbol modulation identification in the WLAN analysis parameters (either Auto Detect, Manual, or Auto from SIG).

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax

```
[SENSe]:WLAN:MODulation:SOURce [AUTO|MANual|SIG}
[SENSe]:WLAN:MODulation:SOURce?
```

Arguments

`AUTO` Auto Detect estimates the modulation from the data symbol IQ content.

`MANual` Manual allows specifying a fixed modulation type regardless of the signal content, using the command `[SENSe]:WLAN:MODulation:MANual`.

`SIG` Auto from SIG sets the modulation as indicated by the embedded SIG preamble symbol format data.

Examples

`SENSE:WLAN:MODULATION:SOURCE MANUAL` enables you to specify a fixed modulation type regardless of the signal content, using the command [SENSe]:WLAN:MODulation:MANual

[SENSe]:WLAN:PILot:TRACking:AMPLitude:STATe

Specifies to use the pilot subcarrier to correct amplitude variations over the packet for WLAN measurements. Note: the default setting is 0 (false).

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:PILot:TRACking:AMPLitude:STATe {0|1|OFF|ON}
[SENSe]:WLAN:PILot:TRACking:AMPLitude:STATe?
```

Arguments

`ON or 1` specifies to use the Amplitude pilot subcarrier to correct amplitude variation over the packets.

`OFF or 0` specifies that the Amplitude pilot subcarrier is off.

Examples

`SENSE:WLAN:PILOT:TRACKING:AMPLITUDE:STATE ON` specifies to use the pilot subcarrier to correct variations.

[SENSe]:WLAN:PILot:TRACking:PHASe:STATe

Specifies to use the pilot subcarrier to correct phase variations over the packet, in WLAN measurements. Note: the default setting is 1 (true).

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax
```
[SENSe]:WLAN:PILot:TRACking:PHASe:STATe {0|1|OFF|ON}
[SENSe]:WLAN:PILot:TRACking:PHASe:STATe?
```

Arguments `ON or 1` specifies to use the pilot subcarrier to correct phase variations over the packet.

`OFF or 0` specifies that the pilot subcarrier is off.

Examples `SENSE:WLAN:PILOT:TRACKING:PHASE:STATE ON` specifies to use the pilot subcarrier to correct phase variations over the packet.

[SENSe]:WLAN:PILot:TRACking:TIMing:STATe

Specifies to use the pilot subcarrier to correct timing variations over the packet, in WLAN measurements. Note: the default setting is 0 (false).

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax
```
[SENSe]:WLAN:PILot:TRACking:TIMing:STATe {0|1|OFF|ON}
[SENSe]:WLAN:PILot:TRACking:TIMing:STATe?
```

Arguments `ON or 1` specifies to use the pilot subcarrier to correct timing variations over the packet.

`OFF or 0` specifies that the pilot subcarrier is off.

Examples `SENSE:WLAN:PILOT:TRACKING:TIMING:STATE ON` specifies to use the pilot subcarrier to correct timing variations over the packet.

[SENSe]:WLAN:RADix

Specifies or queries the format for displaying the symbol values in the WLAN setting preferences (either binary or HEX).

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax

```
[SENSe]:WLAN:RADix {BINary|HEX}
[SENSe]:WLAN:RADix?
```

Arguments `BINary` specifies to display the symbol values in binary format.

`HEX` specifies to display the symbol values in HEX format.

Examples `SENSE:WLAN:RADIX HEX` specifies to display the symbol values in HEX format.

[SENSe]:WLAN:SCARriers

Sets or queries which subcarriers to display in the WLAN view.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax

```
[SENSe]:WLAN:SCARriers {ALL|PILots|DATA|RANGEe|SINGle}
[SENSe]:WLAN:SCARriers?
```

Arguments `ALL` specifies that all subcarriers will be displayed.

`PILot` specifies that only the pilot subcarriers will be displayed.

`DATA` specifies that only the data subcarriers will be displayed.

`RANGe` specifies that the subcarriers within a specified range will be displayed.

`SINGle` specifies that only a single subcarrier will be displayed.

Examples `SENSE:WLAN:SCARRIERS ALL` sets the WLAN view to display all subcarriers.

[SENSe]:WLAN:SCARriers:RANGe:STARt

Sets or queries the start value of the subcarrier range to be displayed in the WLAN view.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax
```
[SENSe]:WLAN:SCARriers:RANGe:STARt <NR1>
[SENSe]:WLAN:SCARriers:RANGe:STARt?
```

Arguments Integer

Examples `SENSE:WLAN:SCARRIERS:RANGE:START 20` sets the subcarrier start range to 20.

[SENSe]:WLAN:SCARriers:RANGe:STOP

Sets or queries the stop value of the subcarrier range to be displayed in the WLAN view.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax
```
[SENSe]:WLAN:SCARriers:RANGe:STOP <NR1>
[SENSe]:WLAN:SCARriers:RANGe:STOP?
```

Arguments Integer

Examples `SENSE:WLAN:SCARRIERS:RANGE:STOP 25` sets the subcarrier stop range to 25.

[SENSe]:WLAN:SCARriers:SINGle:INDex

Specifies or queries the single subcarrier to display in the WLAN view.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax

```
[SENSe]:WLAN:SCARriers:SINGle:INDex <NR1>
[SENSe]:WLAN:SCARriers:SINGle:INDex?
```

Arguments Integer

Examples `SENSE:WLAN:SCARRIERS:SINGLE:INDEX −10` sets the single subcarrier for display to -10.

[SENSe]:WLAN:SCARriers:SPACing

Sets or queries the subcarrier spacing in the WLAN measurements.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax

```
[SENSe]:WLAN:SCARriers:SPACing <NRf>
[SENSe]:WLAN:SCARriers:SPACing?
```

Arguments `NR3` is a floating point number that specifies the subcarrier spacing.

Returns `SENSE:WLAN:SCARRIERs:SPACING 312.5E+3` sets the subcarrier spacing to 312.5 kHz.

[SENSe]:WLAN:SCARriers:SPACing:AUTO

When turned on, automatically sets the subcarrier spacing in the WLAN measurements.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax

```
[SENSe]:WLAN:SCARriers:SPACing:AUTO {0|1|OFF|ON}
[SENSe]:WLAN:SCARriers:SPACing:AUTO?
```

Arguments `OFF or 0` turns the automatic subcarrier spacing off.

`ON or 1` turns the automatic subcarrier spacing on.

Examples `SENSe:WLAN:SCARriers:SPACing:AUTO ON` turns the automatic subcarrier spacing on.

[SENSe]:WLAN:SIGNal:STANdard:TYPE

Sets or queries the WLAN signal standard.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Additionally, the N argument requires Option 24, and the AC argument requires Option 25.

Group Sense commands

Syntax

```
[SENSe]:WLAN:SIGNal:STANdard:TYPE {A|B|G|J|P|N|AC}
[SENSe]:WLAN:SIGNal:STANdard:TYPE?
```

Arguments `A` sets the WLAN signal standard to 802.11a.

`B` sets the WLAN signal standard to 802.11b.

`G` sets the WLAN signal standard to 802.11g.

J sets the WLAN signal standard to 802.11j.

P sets the WLAN signal standard to 802.11p.

N sets the WLAN signal standard to 802.11n. Requires Option 24 in addition to Option 23.

AC sets the WLAN signal standard to 802.11ac. Requires Option 25 in addition to Option 23.

Examples SENSE:WLAN:SIGNAL:STANdARD:TYPE G sets the WLAN signal standard to 802.11g.

[SENSe]:WLAN:SubCARrier:DERotation

Turning on this state allows some displays to show subcarriers with or without the Gamma subcarrier phase rotation removed. Gamma phase rotation is applied to 802.11n and 802.11ac subcarriers in defined subranges depending on the Channel Bandwidth selected (only for Channel Bandwidths of 40 MHz and above).

When derotation is turned off, the rotation is not removed, which provides a direct view of the physical modulation on the channel. With derotation turned on, the rotation is removed, allowing easier decoding of the underlying data content. Only the Constellation and Symbol Table results are affected by this control.

NOTE. *This control is only shown for 802.11n and 802.11ac standard selections for channel bandwidths greater than or equal to 40 MHz. It is not present for all others.*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax

```
[SENSe]:WLAN:SubCARrier:DERotation {0|1|OFF|ON}
[SENSe]:WLAN:SubCARrier:DERotation?
```

Arguments OFF or 0 turns the subcarrier derotation off.

ON or 1 turns the subcarrier derotation on.

Examples SENSE:WLAN:SUBCARRIER:DEROTATION ON turns the subcarrier derotation on.

[SENSe]:WLAN:SUMMary:CLEar:RESults (No Query Form)

Clears the results of the WLAN Summary display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax `[SENSe]:WLAN:SUMMary:CLEar:RESults`

[SENSe]:WLAN:SUMMary:CLEar:RESults (No Query Form)

Clears the results in the WLAN Summary measurement.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax `[SENSe]:WLAN:SUMMary:CLEar:RESults`

Arguments None

Examples `[SENSE]:WLAN:SUMMARY:CLEAR:RESULTS` clears the results in the measurement.

[SENSe]:WLAN:SUMMary:EVM:RMS:AVERage:COUNt

Specifies the maximum number of bursts to average for the WLAN Summary EVM display. The burst average function must be enabled using the command [SENSe]:WLAN:SUMMary:EVM:RMS:AVERage:COUNt:ENABle.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax
```
[SENSe]:WLAN:SUMMary:EVM:RMS:AVERage:COUNt <NR1>
[SENSe]:WLAN:SUMMary:EVM:RMS:AVERage:COUNt?
```

Arguments Integer

Examples `[SENSe]:WLAN:SUMMary:EVM:RMS:AVERage:COUNt 12` sets the maximum number of bursts to average to 12.

[SENSe]:WLAN:SUMMary:EVM:RMS:AVERage:COUNt:ENABle

Enables the burst average function in the WLAN Summary EVM settings. To specify the maximum number of bursts to average, use the command [SENSe]:WLAN:SUMMary:EVM:RMS:AVERage:COUNt.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax
```
[SENSe]:WLAN:SUMMary:EVM:RMS:AVERage:COUNt:ENABle
{0|1|OFF|ON}
[SENSe]:WLAN:SUMMary:EVM:RMS:AVERage:COUNt:ENABle?
```

Arguments `ON or 1` turns the burst average function on.

`OFF or 0` turns the burst average function off.

Examples `SENSE:WLAN:SUMMARY:EVM:RMS:AVERAGE:COUNT:ENABLE ON` turns the burst average function on.

[SENSe]:WLAN:SWAP:IQ

Specifies to swap the I and Q components of a signal in the WLAN analysis parameters. Doing so compensates the input signal for spectral inversion.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax
```
[SENSe]:WLAN:SWAP:IQ {0|1|OFF|ON}
[SENSe]:WLAN:SWAP:IQ?
```

Arguments `OFF or 0` turns the Swap I and Q function off.

`ON or 1` turns the Swap I and Q function on.

Examples `SENSE:WLAN:SWAP:IQ ON` swaps the I and Q components of the signal.

[SENSe]:WLAN:SYMBols|CHIPs

Specifies or queries which symbols or chips are displayed in the WLAN Symbol Table.

Use "CHIPs" for 802.11b and "SYMBols" for all other standards.

The `DAT, PRAM` and `HEAD` enumerations apply to 802.11b only.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax
```
[SENSe]:WLAN:SYMBols|CHIPs {ALL|RANGe|SINGle|DAT|PRAM|HEAD}
[SENSe]:WLAN:SYMBols|CHIPs?
```

Arguments `ALL` specifies that all symbols or chips will be displayed.

`RANGe` specifies that a range of symbols or chips will be displayed.

`SINGle` specifies that single symbols or chips will be displayed.

`DAT` specifies that the Data symbols or chips will be displayed. Applies to 802.11b only.

`PRAM` specifies that Preamble symbols or chips will be displayed. Applies to 802.11b only.

HEAD specifies that Header symbols or chips will be displayed. Applies to 802.11b only.

Examples [SENSE]:WLAN:SYMBOLS ALL specifies that all symbols or chips will be displayed in the WLAN Symbol Table.

[SENSe]:WLAN:SYMBols|CHIPs:MAX

Sets or queries the value for the maximum number of symbols/chips to analyze for the WLAN Symbol Table.

Use "CHIPs" for 802.11b and "SYMBols" for all other standards.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax
```
[SENSe]:WLAN:SYMBols|CHIPs:MAX <NR1>
[SENSe]:WLAN:SYMBols|CHIPs:MAX?
```

Arguments Integer

Examples SENSE:WLAN:SYMBOLS:MAX 50 specifies a maximum value of 50 symbols to analyze for the WLAN Symbol Table.

[SENSe]:WLAN:SYMBols|CHIPs:MAX:STATe

Turns on or off the “maximum data symbols to analyze” feature for the WLAN Symbol Table. To specify the maximum number of symbols/chips to analyze, use the command.

Use "CHIPs" for 802.11b and "SYMBols" for all other standards.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax

```
[SENSe]:WLAN:SYMBols|CHIPs:MAX:STATe {0|1|OFF|ON}
[SENSe]:WLAN:SYMBols|CHIPs:MAX:STATe?
```

Arguments

`ON` or `1` turns on the “maximum data symbols to analyze” feature.

`OFF` or `0` disables the “maximum data symbols to analyze” feature.

Examples

`[SENSE]:WLAN:SYMBOLS:MAX:STATE ON` turns on “maximum data symbols to analyze” feature for the WLAN Symbol Table.

[SENSe]:WLAN:SYMBols|CHIPs:RANGe:COUNt

Sets or queries the number of symbols or chips to display when displaying a range results in the WLAN Symbol Table.

Use "CHIPs" for 802.11b and "SYMBols" for all other standards.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Sense commands

Syntax

```
[SENSe]:WLAN:SYMBols|CHIPs:RANGe:COUNt <NR1>
[SENSe]:WLAN:SYMBols|CHIPs:RANGe:COUNt?
```

Arguments

Integer that represents the number of symbols/chips to displayl

Examples

`[SENSE]:WLAN:SYMBOLS:RANGE:COUNT 4` sets the number of symbols or chips to display to 4 in the WLAN Symbol Table.

[SENSe]:WLAN:SYMBols|CHIPs:RANGe:STARt

Sets or queries the symbols/chips start value when displaying a range of results in the WLAN Symbol Table.

Use "CHIPs" for 802.11b and "SYMBols" for all other standards.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax
```
[SENSe]:WLAN:SYMBols|CHIPs:RANGe:STARt <NR1>
[SENSe]:WLAN:SYMBols|CHIPs:RANGe:STARt?
```

Arguments Integer

Examples `[SENSE]:WLAN:SYMBOLS:RANGE:START 40` sets the start value of the range to 40.

[SENSe]:WLAN:SYMBols|CHIPs:RANGe:STOP

Sets or queries the symbols or chips stop value when displaying a range of results in the WLAN Symbol Table.

Use "CHIPs" for 802.11b and "SYMBols" for all other standards.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax
```
[SENSe]:WLAN:SYMBols|CHIPs:RANGe:STOP <NR1>
[SENSe]:WLAN:SYMBols|CHIPs:RANGe:STOP?
```

Arguments Integer

Examples `[SENSE]:WLAN:SYMBOLS:RANGE:STOP 60` sets the stop value of the range to 60.

[SENSe]:WLAN:SYMBols|CHIPs:SINGle:INDex

Sets or queries the symbol or chip to use when displaying results from a single symbol in the WLAN Symbol Table.

Use "CHIPs" for 802.11b and "SYMBols" for all other standards.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax
```
[SENSe]:WLAN:SYMBols|CHIPs:SINGle:INDex <NR1>
[SENSe]:WLAN:SYMBols|CHIPs:SINGle:INDex?
```

Arguments Integer

Examples `[SENSe]:WLAN:SYMBols|CHIPs:SINGle:INDex 26` specifies to display symbol 26.

[SENSe]:WLAN:UNIT:FREQuency

Specifies or queries whether the displayed frequency units are frequency (Hz) or subcarrier in the WLAN settings.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax
```
[SENSe]:WLAN:UNIT:FREQuency {FREQuency|SCARrier}
[SENSe]:WLAN:UNIT:FREQuency?
```

Arguments `FREQuency` specifies that the displayed frequency units will be Hz.

`SCARrier` specifies that the displayed frequency units will be subcarrier.

Examples `SENSE:WLAN:UNIT:FREQUENCY FREQ` specifies that the frequency units in the display will be Hz.

[SENSe]:WLAN:UNIT:TIME

Specifies or queries whether the displayed time units are seconds, symbols or chips in the WLAN settings. Chips is only a valid option for 802.11b, and Symbols is valid for all other standards.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Sense commands

Syntax
```
[SENSe]:WLAN:UNIT:TIME {SEConds|SYMBols|CHIPs}
[SENSe]:WLAN:UNIT:TIME?
```

Arguments `SEConds` specifies that the displayed time units are to be seconds

`SYMBols` specifies that the displayed time units are to be symbols

`CHIPs` specifies that the displayed time units are to be chips (valid only for 802.11b)

Examples `SENSE:WLAN:UNIT:TIME SYMBOLS` specifies that the displayed time units are to be symbols.

SOURce:TXGain:POWer

Sets or queries the output power level of the tracking generator.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax `SOURce:TXGain:POWer <NRf>`

Arguments

Returns

Examples
```
SOURCE:TXGAIN:POWER
```

*SRE

Sets or queries the value of the Service Request Enable Register (SRER). Refer to Section 3, *Status and Events*, for the register information.

Conditions Measurement views: All

Group IEEE common commands

Syntax
```
*SRE <value>
*SRE?
```

Related Commands *CLS, *ESE, *ESR?, *STB?

Arguments <value> :: <NR1> is a value in the range from 0 to 255.
The binary bits of the SRER are set according to this value.
Using an out-of-range value causes an execution error.

Examples *SRE48 sets binary 00110000 in the SRER's bits.

*SRE? might return 32, indicating that binary value 00100000 has been set in the SRER's bits.

STATus:ACPower:EVENts? (Query Only)

Returns the current events and status conditions for the Channel power and ACPR measurement.

Conditions Measurement views: Channel power and ACPR

Group Status commands

Syntax
```
STATus:ACPower:EVENts?
```

Arguments None

Returns
```
<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}
```

Where
<ecode> :: <NR1> is the error/event code (-32768 to 32767).
<edesc> :: <string> is the description on the error/event.
<einfo> :: <string> is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:ACPOWER:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:{AM|FM|PM}:EVENts? (Query Only)

Returns the current events and status conditions for the AM/FM/PM measurement.

Conditions Measurement views: AM, FM, PM

Group Status commands

Syntax `STATus:{AM|FM|PM}:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:AM:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually in the AM measurement.

STATus:AUDio:SPECtrum:EVENts? (Query Only)

Returns the current events and status conditions for the audio measurement.

Conditions Measurement views: Audio Spectrum

Group Status commands

Syntax `STATus:AUDio:SPECtrum:EVENts?`

Arguments None

Returns <ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where:
<ecode> :: <NR1> is the error/event code (-32768 to 32767).
<edesc> :: <string> is the description on the error/event.
<einfo> :: <string> is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples STATUS:AUDIO:SPECTRUM:EVENTS? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:AUDio:SUMMary:EVENts? (Query Only)

Returns the current events and status conditions for the audio measurement.

Conditions Measurement views: Audio Summary

Group Status commands

Syntax STATus:AUDio:SUMMary:EVENts?

Arguments None

Returns <ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where:
<ecode> :: <NR1> is the error/event code (-32768 to 32767).
<edesc> :: <string> is the description on the error/event.
<einfo> :: <string> is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples STATUS:AUDIO:SUMMARY:EVENTS? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:AVTime:EVENts? (Query Only)

Returns the current events and status conditions for the Amplitude versus Time measurement.

Conditions Measurement views: Amplitude versus Time

Group Status commands

Syntax `STATus:AVTime:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:AVTIME:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:BIBEmissions:EVENts? (Query Only)

Returns the current events and status conditions for the Bluetooth InBand Emission measurement.

Conditions Measurement views: Bluetooth InBand Emission

Group Status commands

Syntax `STATus:BIBEmissions:EVENts?`

Arguments None

Returns <ecode>, “<edesc>[,<einfo>]”{,“<edesc>[,<einfo>]”}

Where:
<ecode> ::= <NR1> is the error/event code (-32768 to 32767).
<edesc> ::= <string> is the description on the error/event.
<einfo> ::= <string> is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:BIBEMISSIONS:EVENTS?` might return 0 indicating no events to report.

STATus:BLUEtooth:CONSte:EVENts? (Query Only)

Returns the current events and status conditions for the Bluetooth Constellation measurement.

Conditions Measurement views: Bluetooth Constellation

Group Status commands

Syntax `STATus:BLUEtooth:CONSte:EVENts?`

Arguments None

Returns <ecode>, “<edesc>[<einfo>]”{, <ecode>,“<edesc>[<einfo>]”}

Where
<ecode> ::= <NR1> is the error/event code (-32768 to 32767).
<edesc> ::= <string> is the description of the error/event.
<einfo> ::= <string> is the additional information of the error/event.

Examples `STATus:BLUEtooth:CONSte:EVENts?` might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:BLUEtooth:EDIAgram:EVENts? (Query Only)

Returns the current events and status conditions for the eye diagram measurement.

Conditions Measurement views: Bluetooth Eye Diagram

Group Status commands

Syntax `STATus:BLUEtooth:EDIAgram:EVENts?`

Arguments None

Returns <ecode>, “<edesc>[<einfo>]”{, <ecode>,“<edesc>[<einfo>]”}

Where

<ecode> ::= <NR1> is the error/event code (-32768 to 32767).

<edesc> ::= <string> is the description of the error/event.

<einfo> ::= <string> is the additional information of the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATus:BLUEtooth:EDIAgram:EVENts?` might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:BLUEtooth:FDVTime:EVENts? (Query Only)

Returns the current events and status conditions for the Frequency Deviation versus Time measurement.

Conditions Measurement views: Frequency Deviation vs. Time

Group Status commands

Syntax `STATus:BLUEtooth:FDVTime:EVENts?`

Arguments None

Returns <ecode>, “<edesc>[,<einfo>]”{,“<edesc>[,<einfo>]”}

Where:

<ecode> ::= <NR1> is the error/event code (-32768 to 32767).

<edesc> ::= <string> is the description on the error/event.

<einfo> ::= <string> is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATus:BLUEtooth:FDVTime:EVENts?` might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:CCDF:EVENts? (Query Only)

Returns the current events and status conditions for the CCDF measurement.

Conditions Measurement views: CCDF

Group Status commands

Syntax `STATus:CCDF:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:CCDF:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:CONSte:EVENts? (Query Only)

Returns the current events and status conditions for the constellation measurement.

Conditions Measurement views: Constellation

Group Status commands

Syntax `STATus:CONSte:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:CONSTE:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:DIQVtime:EVENts? (Query Only)

Returns the current events and status conditions for the Demod I&Q versus Time measurement.

Conditions Measurement views: Demod I&Q versus Time

Group Status commands

Syntax `STATus:DIQVtime:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:DIQVTIME:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:DPX:EVENts? (Query Only)

Returns the current events and status conditions for the DPX, Spectrum measurement.

Conditions Measurement views: DPX, Spectrum

Group Status commands

Syntax `STATus:DPX:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:DPX:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:EDIagram:EVENts? (Query Only)

Returns the current events and status conditions for the eye diagram measurement.

Conditions Measurement views: Eye diagram

Group Status commands

Syntax `STATus:EDIagram:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode>` :: `<NR1>` is the error/event code (-32768 to 32767).
`<edesc>` :: `<string>` is the description on the error/event.
`<einfo>` :: `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:EDIAGRAM:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:EVM:EVENts? (Query Only)

Returns the current events and status conditions for the EVM versus Time measurement.

Conditions Measurement views: EVM versus Time

Group Status commands

Syntax `STATus:EVM:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode>` :: `<NR1>` is the error/event code (-32768 to 32767).
`<edesc>` :: `<string>` is the description on the error/event.
`<einfo>` :: `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:EVM:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:FDVTime:EVENts? (Query Only)

Returns the current events and status conditions for the Frequency deviation versus Time measurement.

Conditions Measurement views: Frequency deviation versus Time

Group Status commands

Syntax `STATus:FDVTime:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:FDVTIME:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:{FSETtling|PSETtling}:EVENts? (Query Only)

Returns the current events and status conditions for the specified settling time measurement.

Conditions Measurement views: Frequency and Phase Settling Time

Group Status commands

Syntax `STATus:{FSETtling|PSETtling}:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:{FSETTLING|PSETTLING}:EVENTS?` might return `2048,"Done saving - restart with Stop, then Run"`, indicating that the acquisition has been completed.

STATus:FVTime:EVENts? (Query Only)

Returns the current events and status conditions for the Frequency versus Time measurement.

Conditions Measurement views: Frequency versus Time

Group Status commands

Syntax `STATus:FVTime:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:FVTIME:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:IQVTime:EVENts? (Query Only)

Returns the current events and status conditions for the RF I&Q versus Time measurement.

Conditions Measurement views: RF I&Q versus Time

Group Status commands

Syntax `STATus:IQVTime:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:IQVTIME:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:LTE:ACLR:EVENts? (Query Only)

Returns the current events and status conditions for the LTE ACLR measurement.

Conditions Measurement view: LTE ACLR

Group Status commands

Syntax `STATus:LTE:ACLR:EVENts?`

Returns <ecode>,”<edesc>[<einfo>]”{,<ecode>,”<edesc>[:<einfo>]”}

Where

<ecode> :: <NR1> is the error/event code (-32768 to 32767).

<edesc> :: <string> is the description of the error/event.

<einfo> :: <string> is the additional information on the error/event.

If there is no error, then response is 0, “No events to report.”

Examples `STATUS:LTE:ACLR:EVENTS?` might return 12006, “Not aligned”,2008,”Not calibrated”,12007,”unaligned data”,12009,”Data from uncalibrated instrument”.

STATus:LTE:CHSPectrum:EVENts? (Query Only)

Returns the current events and status conditions for the LTE Channel Spectrum measurement.

Conditions Measurement view: LTE Channel Spectrum

Group Status commands

Syntax `STATus:LTE:CHSPectrum:EVENts?`

Returns <ecode>,”<edesc>[<einfo>]”{,<ecode>,”<edesc>[:<einfo>]”}

Where

<ecode> :: <NR1> is the error/event code (-32768 to 32767).

<edesc> :: <string> is the description of the error/event.

<einfo> :: <string> is the additional information on the error/event.

If there is no error, then response is 0, “No events to report.”

Examples `STATUS:LTE:CHSPECTRUM:EVENTS?` might return 12006,"Not aligned",2008,"Not calibrated".

STATus:LTE:CONSte:EVENts? (Query Only)

Returns the current events and status condition for the LTE Constellation measurement.

Conditions Measurement view: LTE Constellation

Group Status commands

Syntax `STATus:LTE:CONSte:EVENts?`

Returns <ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode> :: <NR1> is the error/event code (-32768 to 32767).

<edesc> :: <string> is the description of the error/event.

<einfo> :: <string> is the additional information on the error/event.

If there is no error, then response is 0, "No events to report."

Examples `STATUS:LTE:CONSTE:EVENTS?` might return 12006,"Not aligned",2008,"Not calibrated",11090,"Recovery done on PSS/SSS on the center 62, "106 - AnaSynchronization Sequence not found".

STATus:LTE:PVTime:EVENts? (Query Only)

Returns the current events and status conditions for the LTE Power vs Time measurement.

Conditions Measurement view: LTE Power vs Time

Group Status commands

Syntax `STATus:LTE:PVTime:EVENts?`

Returns <ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode> :: <NR1> is the error/event code (-32768 to 32767).

<edesc> :: <string> is the description of the error/event.

<einfo> :: <string> is the additional information on the error/event.

If there is no error, then response is 0, "No events to report."

Examples `STATUS:LTE:PVTIME:EVENTS?` might return 12006,"Not aligned",2008,"Not calibrated".

STATus:MCPower:EVENts? (Query Only)

Returns the current events and status conditions for the MCPR (multi-carrier ACPR) measurement.

Conditions Measurement views: MCPR

Group Status commands

Syntax `STATus:MCPower:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:MCPOWER:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:MERRor:EVENts? (Query Only)

Returns the current events and status conditions for the Magnitude error versus Time measurement.

Conditions Measurement views: Magnitude error versus Time

Group Status commands

Syntax `STATus:MERRor:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:MERROR:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:OBWidth:EVENts? (Query Only)

Returns the current events and status conditions for the Occupied Bandwidth (OBW) measurement.

Conditions Measurement views: Occupied Bandwidth

Group Status commands

Syntax `STATus:OBWidth:EVENts?`

Arguments None

Returns <ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:OBWIDTH:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:OFDM:CONSte:EVENts? (Query Only)

Returns the current events and status condition for the OFDM Constellation measurement.

Conditions Measurement views: OFDM

Group Status commands

Syntax `STATus:OFDM:CONSte:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:OFDM:CONSTE:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:OFDM:EVM:EVENts? (Query Only)

Returns the current events and status condition for the OFDM EVM (Error Vector Magnitude) versus Time measurement.

Conditions Measurement views: OFDM

Group Status commands

Syntax `STATus:OFDM:EVM:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:OFDM:EVM:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:OFDM:MERRor:EVENts? (Query Only)

Returns the current events and status condition for the OFDM Magnitude versus Time measurement.

Conditions Measurement views: OFDM

Group Status commands

Syntax `STATus:OFDM:MERRor:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:OFDM:MERROR:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:OFDM:PERRor:EVENts? (Query Only)

Returns the current events and status condition for the OFDM Phase error versus Time measurement.

Conditions Measurement views: OFDM

Group Status commands

Syntax `STATus:OFDM:PERRor:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:OFDM:PERROR:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:OFDM:POWer:EVENts? (Query Only)

Returns the current events and status condition for the OFDM Power measurement.

Conditions Measurement views: OFDM

Group Status commands

Syntax `STATus:OFDM:POWer:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
<ecode> :: <NR1> is the error/event code (-32768 to 32767).
<edesc> :: <string> is the description on the error/event.
<einfo> :: <string> is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples STATUS:OFDM:POWER:EVENTS? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:OFDM:STABle:EVENts? (Query Only)

Returns the current events and status condition for the OFDM Symbol table measurement.

Conditions Measurement views: OFDM

Group Status commands

Syntax STATus:OFDM:STABle:EVENts?

Arguments None

Returns <ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where
<ecode> :: <NR1> is the error/event code (-32768 to 32767).
<edesc> :: <string> is the description on the error/event.
<einfo> :: <string> is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples STATUS:OFDM:STABLE:EVENTS? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:OPERation:CONDition? (Query Only)

Returns the contents of the Operation Condition Register (OCR).

Conditions Measurement views: All

Group Status commands

Syntax `STATus:OPERation:CONDition?`

Arguments None

Returns <NR1> is a decimal number showing the contents of the OCR.

Examples `STATUS:OPERATION:CONDITION?` might return `16`, showing that the bits in the OCR have the binary value 0000000000010000, which means the analyzer is in measurement.

STATus:OPERation:ENABle

Sets or queries the enable mask of the Operation Enable Register (OENR) which allows true conditions in the Operation Event Register to be reported in the summary bit.

Conditions Measurement views: All

Group Status commands

Syntax
```
STATus:OPERation:ENABle <bit_value>
STATus:OPERation:ENABle?
```

Arguments <bit_value> :: <NR1> is the enable mask of the OENR. Range: 0 to 65535.

Returns <NR1> is a decimal number showing the contents of the OENR.
Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples `STATUS:OPERATION:ENABLE1` enables the ALIGning bit.

`STATUS:OPERATION:ENABLE?` might return `1`, showing that the bits in the OENR have the binary value 00000000 00000001, which means that the ALIGning bit is valid.

STATus:OPERation[:EVENt]? (Query Only)

Returns the contents of the Operation Event Register (OEVR). Reading the OEVR clears it.

Conditions Measurement views: All

Group Status commands

Syntax

```
STATus:OPERation[:EVENt]?
```

Arguments None

Returns <NR1> is a decimal number showing the contents of the OEVR.

Examples STATUS:OPERATION:EVENT? might return 1, showing that the bits in the OEVR have the binary value 00000000 00000001, which means that the ALIGning bit is set.

STATus:OPERation:NTRansition

Sets or queries the negative transition filter value of the Operation Transition Register (OTR).

Conditions Measurement views: All

Group Status commands

Syntax

```
STATus:OPERation:NTRansition <bit_value>
STATus:OPERation:NTRansition?
```

Arguments <bit_value> :: <NR1> is the negative transition filter value. Range: 0 to 65535.

Returns <NR1> is a decimal number showing the contents of the OTR. Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples `STATUS:OPERATION:NTRANSITION#H0011` sets the negative transition filter value to #H0011.

`STATUS:OPERATION:NTRANSITION?` might return `17`.

STATus:OPERation:PTRansition

Sets or queries the positive transition filter value of the Operation Transition Register (OTR).

Conditions Measurement views: All

Group Status commands

Syntax
```
STATus:OPERation:PTRansition <bit_value>
STATus:OPERation:PTRansition?
```

Arguments `<bit_value> ::` `<NR1>` is the positive transition filter value. Range: 0 to 65535.

Returns `<NR1>` is a decimal number showing the contents of the OTR.
Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples `STATUS:OPERATION:PTRANSITION0` sets the positive transition filter value to 0.

`STATUS:OPERATION:PTRANSITION?` might return `0`.

STATus:P25:EDIagram:EVENts? (Query Only)

Returns the current events and status conditions for the P25 Eye Diagram display.

Conditions Measurement view: P25 Eye Diagram

Group Status commands

Syntax
```
STATus:P25:EDIagram:EVENts?
```

Returns <ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where:

<ecode> :: <NR1> is the error/event code (-32768 to 32767).

<edesc> :: <string> is the description of the error/event.

<einfo> :: <string> is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:P25:EDIAGRAM:EVENTS?` might return 12026,"Acq Sampling Parma's: Manual control", indicating that the sampling parameters are controlled Manually.

STATus:PERRor:EVENts? (Query Only)

Returns the current events and status conditions for the Phase error versus Time measurement.

Conditions Measurement views: Phase error versus Time

Group Status commands

Syntax `STATus:PERRor:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:PERROR:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:PHVTime:EVENts? (Query Only)

Returns the current events and status conditions for the Phase versus Time measurement.

Conditions Measurement views: Phase versus Time

Group Status commands

Syntax `STATus:PHVTime:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:PHVTIME:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:PNOise:EVENts? (Query Only)

Returns the current events and status conditions for the phase noise measurement.

Conditions Measurement views: Spurious

Group Status commands

Syntax `STATus:PNOise:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:PNOISE:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:PRESet (No Query Form)

Presets the SCPI enable registers and transition registers.

Conditions Measurement views: All

Group Status commands

Syntax `STATus:PRESet`

Arguments None

Examples `STATUS:PRESET` presets the SCPI enable registers and transition registers.

STATus:PULSe:RESult:EVENts? (Query Only)

Returns the current events and status conditions for the pulse table measurement.

Conditions Measurement views: Pulse table

Group Status commands

Syntax `STATus:PULSe:RESult:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.

<einfo> :: <string> is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples STATUS:PULSE:RESULT:EVENTS? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:PULSe:STATistics:EVENts? (Query Only)

Returns the current events and status conditions for the pulse statistics measurement.

Conditions Measurement views: Pulse statistics

Group Status commands

Syntax STATus:PULSe:STATistics:EVENts?

Arguments None

Returns <ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where
<ecode> :: <NR1> is the error/event code (-32768 to 32767).
<edesc> :: <string> is the description on the error/event.
<einfo> :: <string> is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples STATUS:PULSE:STATISTICS:EVENTS? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:PULSe:TRACe:EVENts? (Query Only)

Returns the current events and status conditions for the pulse trace measurement.

Conditions Measurement views: Pulse trace

Group Status commands

Syntax `STATus:PULSe:TRACe:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:PULSE:TRACE:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:QUEStionable:CALibration:CONDition? (Query Only)

Returns the contents of the questionable calibration condition register.

Conditions Measurement views: All

Group Status commands

Syntax `STATus:QUEStionable:CALibration:CONDition?`

Arguments None

Returns `<NR1>` is a decimal number showing the contents of the questionable calibration condition register.

Examples `STATUS:QUESTIONABLE:CALIBRATION:CONDITION?` might return `16384`, showing that the bits in the questionable calibration condition register have the binary value 01000000 00000000, which means the Alignment Needed bit is set.

STATus:QUEStionable:CALibration:ENABle

Sets or queries the enable mask of the questionable calibration enable register which allows true conditions in the questionable calibration event register to be reported in the summary bit.

Conditions Measurement views: All

Group Status commands

Syntax
```
STATus:QUEStionable:CALibration:ENABle <bit_value>
STATus:QUEStionable:CALibration:ENABle?
```

Arguments <bit_value> :: <NR1> is the enable mask of the questionable calibration enable register. Range: 0 to 65535.

Returns <NR1> is a decimal number showing the contents of the questionable calibration enable register. Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples STATUS:QUESTIONABLE:CALIBRATION:ENABLE16384 enables the Alignment Needed bit.

STATUS:QUESTIONABLE:CALIBRATION:ENABLE? might return 16384, showing that the bits in the questionable calibration enable register have the binary value 01000000 00000000, which means that the Calibration Summary bit is valid.

STATus:QUEStionable:CALibration[:EVENt]? (Query Only)

Returns the contents of the questionable calibration event register. Reading the register clears it.

Conditions Measurement views: All

Group Status commands

Syntax
```
STATus:QUEStionable:CALibration[:EVENt]?
```

Arguments None

Returns <NR1> is a decimal number showing the contents of the questionable calibration event register.

Examples STATUS:QUESTIONABLE:CALIBRATION:EVENT? might return 16384, showing that the bits in the questionable calibration event register have the binary value 01000000 00000000, which means that the Calibration Summary bit is set.

STATus:QUEStionable:CALibration:NTRansition

Sets or queries the negative transition filter value of the questionable calibration transition register.

Conditions Measurement views: All

Group Status commands

Syntax
```
STATus:QUEStionable:CALibration:NTRansition <bit_value>
STATus:QUEStionable:CALibration:NTRansition?
```

Arguments <bit_value> :: <NR1> is the negative transition filter value. Range: 0 to 65535.

Returns <NR1> is a decimal number showing the contents of the questionable calibration transition register. Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples STATUS:QUESTIONABLE:CALIBRATION:NTRANSITION#H4000 sets the negative transition filter value to #H4000.

STATUS:QUESTIONABLE:CALIBRATION:NTRANSITION? might return 16384.

STATus:QUEStionable:CALibration:PTRansition

Sets or queries the positive transition filter value of the questionable calibration transition register.

Conditions Measurement views: All

Group Status commands

Syntax

```
STATus:QUEStionable:CALibration:PTRansition <bit_value>
STATus:QUEStionable:CALibration:PTRansition?
```

Arguments

<bit_value> :: <NR1> is the positive transition filter value. Range: 0 to 65535.

Returns

<NR1> is a decimal number showing the contents of the questionable calibration transition register. Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples

STATUS:QUESTIONABLE:CALIBRATION:PTRANSITION0 sets the positive transition filter value to 0.

STATUS:QUESTIONABLE:CALIBRATION:PTRANSITION? might return 0.

STATus:QUEStionable:CONDition? (Query Only)

Returns the contents of the Questionable Condition Register (QCR).

Conditions

Measurement views: All

Group

Status commands

Syntax

```
STATus:QUEStionable:CONDition?
```

Arguments

None

Returns

<NR1> is a decimal number showing the contents of the QCR.

Examples

STATUS:QUESTIONABLE:CONDITION? might return 256, showing that the bits in the QCR have the binary value 00000001 00000000, which means the Calibration Summary bit is set.

STATus:QUEStionable:ENABle

Sets or queries the enable mask of the Questionable Enable Register (QENR) which allows true conditions in the Questionable Event Register to be reported in the summary bit.

Conditions Measurement views: All

Group Status commands

Syntax
```
STATus:QUEStionable:ENABle <bit_value>
STATus:QUEStionable:ENABle?
```

Arguments <bit_value> :: <NR1> is the enable mask of QENR. Range: 0 to 65535.

Returns <NR1> is a decimal number showing the contents of the QENR. Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples STATUS:QUESTIONABLE:ENABLE256 enables the Calibration Summary bit.

STATUS:QUESTIONABLE:ENABLE? might return 256, showing that the bits in the QENR have the binary value 00000001 00000000, which means that the Calibration Summary bit is valid.

STATus:QUEStionable[:EVENt]? (Query Only)

Returns the contents of the Questionable Event Register (QEVR). Reading the QEVR clears it.

Conditions Measurement views: All

Group Status commands

Syntax
```
STATus:QUEStionable[:EVENt]?
```

Arguments None

Returns <NR1> is a decimal number showing the contents of the QEVR.

Examples STATUS:QUESTIONABLE:EVENT? might return 256, showing that the bits in the QEVR have the binary value 00000001 00000000, which means that the Calibration Summary bit is set.

STATus:QUEStionable:FREQuency:CONDition? (Query Only)

Returns the contents of the questionable frequency condition register.

Conditions Measurement views: All

Group Status commands

Syntax `STATus:QUEStionable:FREQuency:CONDition?`

Arguments None

Returns <NR1> is a decimal number showing the contents of the questionable frequency condition register.

Examples `STATUS:QUESTIONABLE:FREQUENCY:CONDITION?` might return `512`, showing that the bits in the questionable frequency condition register have the binary value 00000010 00000000, which means the Locked To External Ref bit is set.

STATus:QUEStionable:FREQuency:ENABle

Sets or queries the enable mask of the questionable frequency enable register which allows true conditions in the questionable frequency event register to be reported in the summary bit.

Conditions Measurement views: All

Group Status commands

Syntax
```
STATus:QUEStionable:FREQuency:ENABle <bit_value>
STATus:QUEStionable:FREQuency:ENABle?
```

Arguments <bit_value> :: <NR1> is the enable mask of the questionable frequency enable register. Range: 0 to 65535.

Returns <NR1> is a decimal number showing the contents of the questionable frequency enable register. Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples

`STATUS:QUESTIONABLE:FREQUENCY:ENABLE512` enables the Locked To External Ref bit.

`STATUS:QUESTIONABLE:FREQUENCY:ENABLE?` might return `512`, showing that the bits in the questionable calibration enable register have the binary value 00000010 00000000, which means that the Locked To External Ref bit is valid.

STATus:QUEStionable:FREQuency[:EVENt]? (Query Only)

Returns the contents of the questionable frequency event register. Reading the register clears it.

Conditions

Measurement views: All

Group

Status commands

Syntax

```
STATus:QUEStionable:FREQuency[:EVENt]?
```

Arguments

None

Returns

`<NR1>` is a decimal number showing the contents of the questionable frequency event register.

Examples

`STATUS:QUESTIONABLE:FREQUENCY:EVENT?` might return `512`, showing that the bits in the questionable frequency event register have the binary value 00000010 00000000, which means that the Locked To External Ref bit is set.

STATus:QUEStionable:FREQuency:NTRansition

Sets or queries the negative transition filter value of the questionable frequency transition register.

Conditions

Measurement views: All

Group

Status commands

Syntax

```
STATus:QUEStionable:FREQuency:NTRansition <bit_value>
STATus:QUEStionable:FREQuency:NTRansition?
```

Arguments <bit_value> :: <NR1> is the negative transition filter value. Range: 0 to 65535.

Returns <NR1> is a decimal number showing the contents of the questionable frequency transition register. Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples STATUS:QUESTIONABLE:FREQUENCY:NTRANSITION#H0200 sets the negative transition filter value to #H0200.

STATUS:QUESTIONABLE:FREQUENCY:NTRANSITION? might return 512.

STATus:QUEStionable:FREQuency:PTRansition

Sets or queries the positive transition filter value of the questionable frequency transition register.

Conditions Measurement views: All

Group Status commands

Syntax
```
STATus:QUEStionable:FREQuency:PTRansition <bit_value>
STATus:QUEStionable:FREQuency:PTRansition?
```

Arguments <bit_value> :: <NR1> is the positive transition filter value. Range: 0 to 65535.

Returns <NR1> is a decimal number showing the contents of the questionable frequency transition register. Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples STATUS:QUESTIONABLE:FREQUENCY:PTRANSITION0 sets the positive transition filter value to 0.

STATUS:QUESTIONABLE:FREQUENCY:PTRANSITION? might return 0.

STATus:QUEStionable:NTRansition

Sets or queries the negative transition filter value of the Questionable Transition Register (QTR).

Conditions Measurement views: All

Group Status commands

Syntax
```
STATus:QUEStionable:NTRansition <bit_value>
STATus:QUEStionable:NTRansition?
```

Arguments <bit_value> :: <NR1> is the negative transition filter value. Range: 0 to 65535.

Returns <NR1> is a decimal number showing the contents of the QTR.
Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples STATUS:QUESTIONABLE:NTRANSITION#H0020 sets the negative transition filter value to #H0020.

STATUS:QUESTIONABLE:NTRANSITION? might return 32.

STATus:QUEStionable:PTRansition

Sets or queries the positive transition filter value of the Questionable Transition Register (QTR).

Conditions Measurement views: All

Group Status commands

Syntax
```
STATus:QUEStionable:PTRansition <bit_value>
STATus:QUEStionable:PTRansition?
```

Arguments <bit_value> :: <NR1> is the positive transition filter value. Range: 0 to 65535.

Returns <NR1> is a decimal number showing the contents of the QTR.
Range: 0 to 32767 (The most-significant bit cannot be set true.)

Examples STATUS:QUESTIONABLE:PTRANSITION0 sets the positive transition filter value to 0.

STATUS:QUESTIONABLE:PTRANSITION? might return 0.

STATus:SEM:EVENts? (Query Only)

Returns the current events and status conditions for the Spectral Emissions Mask measurement.

Conditions Measurement views: Spectral Emissions Mask

Group Status commands

Syntax `STATus:SEM:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:SEM:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:SGRAM:EVENts? (Query Only)

Returns the current events and status conditions for the spectrogram measurement.

Conditions Measurement views: Spectrogram

Group Status commands

Syntax `STATus:SGRAM:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
<ecode> :: <NR1> is the error/event code (-32768 to 32767).
<edesc> :: <string> is the description on the error/event.
<einfo> :: <string> is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples STATUS:SGRAM:EVENTS? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:SPECtrum:EVENts? (Query Only)

Returns the current events and status conditions and status conditions for the spectrum measurement.

Conditions Measurement views: Spectrum

Group Status commands

Syntax STATus:SPECtrum:EVENts?

Arguments None

Returns <ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where
<ecode> :: <NR1> is the error/event code (-32768 to 32767).
<edesc> :: <string> is the description on the error/event.
<einfo> :: <string> is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples STATUS:SPECTRUM:EVENTS? might return 12026,"Acq Sampling Params: manual control", indicating that the sampling parameters are controlled manually.

STATus:SPURious:EVENts? (Query Only)

Returns the current events and status conditions for the Spurious measurement.

Conditions Measurement views: Spurious

Group Status commands

Syntax `STATus:SPURious:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:SPURIOUS:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:SQUality:EVENts? (Query Only)

Returns the current events and status conditions for the signal quality measurement.

Conditions Measurement views: Signal quality

Group Status commands

Syntax `STATus:SQUality:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:SQUALITY:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:TDIagram:EVENts? (Query Only)

Returns the current events and status conditions for the trellis diagram measurement.

Conditions Measurement views: Trellis diagram

Group Status commands

Syntax `STATus:TDIagram:EVENts?`

Arguments None

Returns `<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:TDIAGRAM:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:TXGain:EVENts? (Query Only)

Queries events for the Transmission Gain display.

Conditions Measurement view: Transmission Gain

Group Display group

Syntax `STATus:TXGain:EVENts?`

Returns

Examples `STATUS:TXGAIN:EVENTS?`

STATus:WLAN:CONSte:EVENts? (Query Only)

Returns the current instrument status concatenated with WLAN constellation-specific event information.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Status commands

Syntax `STATus:WLAN:CONSte:EVENts?`

Returns ASCII string representation of instrument status as follows:

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode> :: <NR1> is the error/event code (-32768 to 32767).

<edesc> :: <string> is the description on the error/event.

<einfo> :: <string> is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:WLAN:CONSTE:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:WLAN:CRESponse:EVENts? (Query Only)

Returns the current events and status conditions for the WLAN Channel Response display.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Status commands

Syntax `STATus:WLAN:CRESponse:EVENts?`

Returns <ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode> :: <NR1> is the error/event code (-32768 to 32767).

<edesc> :: <string> is the description on the error/event.

<einfo> :: <string> is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:WLAN:CRES:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:WLAN:EVM:EVENts? (Query Only)

Returns the current instrument status concatenated with WLAN EVM-specific event information.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Status commands

Syntax `STATus:WLAN:EVM:EVENts?`

Returns ASCII string representation of instrument status as follows:

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode> :: <NR1> is the error/event code (-32768 to 32767).

<edesc> :: <string> is the description on the error/event.

<einfo> :: <string> is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:WLAN:EVM:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:WLAN:MERRor:EVENts? (Query Only)

Returns the current instrument status concatenated with WLAN Magnitude Error-specific event information.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Status commands

Syntax `STATus:WLAN:MERRor:EVENts?`

Returns <ecode,"<edesc[<einfo]"{,<ecode,"<edesc[:<einfo]"}

Where:

<ecode:: <NR1> is the error/event code (-32768 to 32767).

<edesc:: <string> is the description on the error/event.

<einfo:: <string> is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:WLAN:MERR:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:WLAN:PERRor:EVENts? (Query Only)

Returns instrument status concatenated with WLAN Phase Error-specific event information.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Status commands

Syntax `STATus:WLAN:PERRor:EVENts?`

Returns ASCII string representation of instrument status as follows:

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode> :: <NR1> is the error/event code (-32768 to 32767).

<edesc> :: <string> is the description on the error/event.

<einfo> :: <string> is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:WLAN:PERR:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:WLAN:PVTime:EVENts? (Query Only)

Returns the current instrument status concatenated with WLAN Power vs. Time-specific event information.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Status commands

Syntax `STATus:WLAN:PVTime:EVENts?`

Returns ASCII string representation of instrument status as follows:

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode> :: <NR1> is the error/event code (-32768 to 32767).

<edesc> :: <string> is the description on the error/event.

<einfo> :: <string> is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:WLAN:PVT:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:WLAN:STABle:EVENts? (Query Only)

Returns instrument status concatenated with WLAN Symbol Table-specific event information.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Status commands

Syntax `STATus:WLAN:STABle:EVENts?`

Returns ASCII string representation of instrument status as follows:

<ecode>,"<edesc>[<einfo>]"{,<ecode>,"<edesc>[:<einfo>]"}

Where

<ecode> :: <NR1> is the error/event code (-32768 to 32767).

<edesc> :: <string> is the description on the error/event.

<einfo> :: <string> is the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

Examples `STATUS:WLAN:STABLE:EVENTS?` might return `12026,"Acq Sampling Params: manual control"`, indicating that the sampling parameters are controlled manually.

STATus:WLAN:SUMMary:EVENts? (Query Only)

Returns the current events and status conditions for the WLAN summary measurement.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Status commands

Syntax `STATus:WLAN:SUMMary:EVENts?`

Arguments None

Returns <ecode>, “<edesc>[<einfo>]” {, <ecode>, “<edesc>[<einfo>]”}

Where

<ecode> :: <NR1> the error/event code (-32768 to 32767).
<edesc> :: <string> the description on the error/event.
<einfo> :: <string> the additional information on the error/event.

If there is no error, the response is 0, "No events to report".

*STB? (Query Only)

Returns the contents of the Status Byte Register (SBR) in the status/event reporting structure using the Master Summary Status (MSS) bit. Refer to Section3, *Status and Events*, for the register information.

Conditions Measurement views: All

Group IEEE common commands

Syntax `*STB?`

Related Commands *CLS, *ESE, *ESR?, *SRE

Arguments None

Returns <NR1> representing the contents of the SBR as a decimal number.

Examples *STB? might return 96, indicating that the SBR contains binary 0110 0000.

SYSTem:BATTery:CCYCle? (Query Only)

Queries the battery charge cycle.

Conditions Requires an RSA500A series instrument with a battery installed.

Group System group

Syntax SYSTem:BATTery:CCYCle?

Returns Returns the number of charge cycles of the battery.

Examples SYSTEM:BATTERY:CCYCLE? might return 10, indicating that the battery has been charged 10 times.

SYSTem:BATTery:DATE? (Query Only)

Queries the battery manufactured date.

Conditions Requires an RSA500A series instrument with a battery installed.

Group System group

Syntax SYSTem:BATTery:DATE?

Returns Returns the date string as yyyy-mm-dd.

Examples SYSTEM:BATTERY:DATE? might return "2015-10-05", indicating the battery manufactured date is October 5, 2015.

SYSTem:BATTery:RCHarge? (Query Only)

Queries the relative charge remaining.

Conditions Requires an RSA500A series or RSA600A series instrument with a battery installed.

Group System group

Syntax `SYSTem:BATTery:RCHarge?`

Returns Returns the relative charge remaining in percent.

Examples `SYSTEM:BATTERY:RCHARGE?` might return 50, indicating the relative charge remaining is 50%.

SYSTem:BATTery:SERial? (Query Only)

Queries the battery serial number.

Conditions Requires an RSA500A series instrument with a battery installed.

Group System group

Syntax `SYSTem:BATTery:SERial?`

Returns Returns the battery serial number.

Examples `SYSTEM:BATTERY:SERIAL?` might return 1234, indicating the serial number is 1234.

SYSTem:BATTery:STATus? (Query Only)

Queries the battery status.

Conditions Requires an RSA500A series instrument with a battery installed.

Group System group

Syntax `SYSTem:BATTery:STATus??`

Returns ACFull means AC present and fully charged

ACCHarging means AC present and charging

ACNBattery means AC present and no battery

FULL means battery fully charged

EMPTy means battery fully discharged

DISCharging battery discharging

ERRor battery error

Examples `SYSTEM:BATTERY:STATUS??` might return ACNB, indicating no battery is installed.

SYSTem:BATTery:TEMPerature? (Query Only)

Queries the battery temperature.

Conditions Requires an RSA500A series instrument with a battery installed.

Group System group

Syntax `SYSTem:BATTery:TEMPerature?`

Returns Returns the battery temperature in degrees Celsius.

Examples `SYSTEM:BATTERY:TEMPERATURE?` might return 15, indicating that the battery temperature is 15 °C.

SYSTem:COMMunicate:GPIB[:SELF]:ADDRess

Sets or queries the GPIB address of the instrument.

Conditions Measurement views: All

Group System commands

Syntax
```
SYSTem:COMMunicate:GPIB[:SELF]:ADDRess <value>
SYSTem:COMMunicate:GPIB[:SELF]:ADDRess?
```

Arguments <value> :: <NR1> specifies the GPIB address of the instrument.
*RST has no effect on the value.

Examples SYSTEM:COMMUNICATE:GPIB:SELF:ADDRESS 18 sets the GPIB address to 18.

SYSTem:DATE

Sets or queries the date (year, month, and day). This command is equivalent to the date setting through the Windows Control Panel.

Conditions Measurement views: All

Group System commands

Syntax
```
SYSTem:DATE <year>,<month>,<day>
SYSTem:DATE?
```

Related Commands SYSTem:TIME

Arguments <year> :: <NRf> specifies the year (4 digits). Range: 2000 to 2099.
<month> :: <NRf> specifies the month. Range: 1 (January) to 12 (December).
<day> :: <NRf> specifies the day. Range: 1 to 31.
These values are rounded to the nearest integer.

*RST has no effect on the settings.

Examples SYSTEM:DATE2008,3,19 sets the internal calendar to March 19, 2008.

SYSTem:ERRor:ALL? (Query Only)

Queries the error/event queue for all the unread items and removes them from the queue. The response is a comma separated list of number, string pairs in FIFO order. For details of the error messages, refer to (See Table 3-16.)

Conditions Measurement views: All

Group System commands

Syntax `SYSTem:ERRor:ALL?`

Arguments None

Returns `<ecode>,"<edesc>[;<einfo>]"{,<ecode>,"<edesc>[;<einfo>]"}`

Where
`<ecode> ::` `<NR1>` is the error/event code (-32768 to 32767).
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the detail of the error/event.

If the queue is empty, the response is 0, "No error; Queue empty - No events to report".

Examples `SYSTEM:ERROR:ALL?` might return `-130, "Suffix error; Unrecognized suffix, INPutMLEVel -10dB",` , indicating that the unit of the reference level is improper.

SYSTem:ERRor:CODE:ALL? (Query Only)

Queries the error/event queue for all the unread items and removes them from the queue. The response returns a comma separated list of only the error/event code numbers in FIFO order. For details of the error messages, refer to (See Table 3-16.)

Conditions Measurement views: All

Group System commands

Syntax `SYSTem:ERRor:CODE:ALL?`

Arguments None

Returns `<ecode>{,<ecode>}`

Where
`<ecode> ::` `<NR1>` is the error/event code, ranging from -32768 to 32767.

If the queue is empty, the response is 0.

Examples `SYSTEM:ERROR:CODE:ALL?` might return `-101,-108` of the error codes.

SYSTem:ERRor:CODE[:NEXT]? (Query Only)

Queries the error/event queue for the next item and removes it from the queue. The response returns only the error/event code number omitting the string. Except for the shortened response, the query operates identically to SYSTem:ERRor[:NEXT]?. For details of the error messages, refer to (See Table 3-16.)

Conditions Measurement views: All

Group System commands

Syntax `SYSTem:ERRor:CODE[:NEXT]?`

Arguments None

Returns `<ecode> ::` `<NR1>` is the error/event code, ranging from -32768 to 32767.

Examples `SYSTEM:ERROR:CODE:NEXT?` might return `-101` of the error code.

SYSTem:ERRor:COUNt? (Query Only)

Queries the error/event queue for the number of unread items. As errors and events may occur at any time, more items may be present in the queue at the time it is actually read.

Conditions Measurement views: All

Group System commands

Syntax `SYSTem:ERRor:COUNt?`

Arguments None

Returns `<enum> ::` `<NR1>` is the number of errors/events.

If the queue is empty, the response is 0.

Examples `SYSTEM:ERROR:COUNT?` might return 2, indicating that the error/event queue contains two of unread errors/events.

SYSTem:ERRor[:NEXT]? (Query Only)

Queries the error/event queue for the next item and removes it from the queue. The response returns the full queue item consisting of an integer and a string. For details of the error messages, refer to (See Table 3-16.)

Conditions Measurement views: All

Group System commands

Syntax `SYSTem:ERRor[:NEXT]?`

Arguments None

Returns `<ecode>,"<edesc>[;<einfo>]"`

Where
`<ecode> ::` `<NR1>` is the error/event code, ranging from -32768 to 32767.
`<edesc> ::` `<string>` is the description on the error/event.
`<einfo> ::` `<string>` is the additional information on the error/event.

Examples `SYSTEM:ERROR:NEXT?` might return `-130, "Suffix error; Unrecognized suffix, INPutMLEVel -10dB"`, indicating that the unit is improper.

SYSTem:KLOCk

Enables or disables the local lockout operation.

Conditions Measurement views: All

Group System commands

Syntax

```
SYSTem:KLOCk { OFF | ON | 0 | 1 }
SYSTem:KLOCk?
```

Arguments

OFF or 0 disables the local lockout operation.

ON or 1 enables the local lockout operation.

Examples

`SYSTEM:KLOCK ON` enables the local lockout operation.

SYSTem:OPTions? (Query Only)

Queries the options installed in the analyzer. This command is equivalent to the IEEE common command *OPT?.

Conditions

Measurement views: All

Group

System commands

Syntax

```
SYSTem:OPTions?
```

Arguments

None

Returns

`<option> :: <string>` contains the comma-separated option numbers.

Examples

`SYSTEM:OPTIONS?` might return `"01,02,20"`, indicating that Option 01, 02, and 20 are currently installed in the analyzer.

SYSTem:PRESet (No Query Form)

Restores the analyzer to the defaults. This command is equivalent to the **Preset** key on the front panel.

Conditions

Measurement views: All

Group

System commands

Syntax

```
SYSTem:PRESet
```

Arguments None

Examples `SYSTEM:PRESET` restores the analyzer to the defaults.

SYSTem:PRESet:APPLication (No Query Form)

Restores the analyzer to the defaults for the application preset type.

Conditions Measurement views: All

Group System commands

Syntax
```
SYSTem:PRESet:APPLication { TIMFreq | SPECtrum | MODanalysis
| PULSe | SPURious }
```

Arguments `TIMFreq` displays the time-frequency analysis for the spectrum and spectrogram.

`SPECtrum` displays the spectrum analysis.

`MODanalysis` displays the symbol table, constellation, and signal quality for the DPX, Spectrum.

`PULSe` displays the pulse trace, pulse table, and time overview for the DPX, Spectrum.

`SPURious` displays the spurious display for the RF measurements.

Examples `SYSTEM:PRESET:APPLICATION TIMFreq` displays the time-frequency analysis for the spectrum and spectrogram.

SYSTem:PRESet:APPLication:ACTion

Sets or queries the preset action for the application preset type.

Conditions Measurement views: All

Group System commands

Syntax
```
SYSTem:PRESet:APPLication:ACTion { RECall | SHOW }
SYSTem:PRESet:APPLication:ACTion?
```

Arguments `RECall` recalls the selected preset for the application preset type.

`SHOW` lists the available presets for the application preset type.

Examples `SYSTEM:PRESET:APPLICATION:ACTION SHOW` lists the available presets for the application preset type.

SYSTem:PRESet:APPLication:SELected

Sets or queries the presets for the application preset type.

Conditions Measurement views: All

Group System commands

Syntax
```
SYSTem:PRESet:APPLication:SELected { TIMFreq | SPECtrum |
MODanalysis | PNOise | PULSe | SPURious }
SYSTem:PRESet:APPLication:SELected?
```

Arguments `TIMFreq` sets the Application preset to Time-Frequency analysis.

`SPECtrum` sets the Application preset to Spectrum Analysis.

`MODanalysis` sets the Application preset to Modulation Analysis.

`PNOise` sets the Application preset to Phase Noise.

`PULSe` sets the preset to Pulse Analysis.

`SPURious` sets the Application preset to the Spurious Analysis Multi Zone (9 k to 1 GHz).

Examples `SYSTEM:PRESET:APPLICATION:SELECTED TIMFreq` sets the presets to the time-frequency analysis.

SYSTem:PRESet:BLUEtooth:STANdard

Sets or queries the standard, setup, Retaining Current Reflevel, and Retaining Center Frequency in the Bluetooth standard preset.

Conditions Measurement views: Launch BT Application

Group System commands

Syntax
```
SYSTem:PRESet:BLUEtooth:STANdard {BR|LE|EDR}, <test_setup>,
<Retaining_Current_Reflevel>, <Retaining_Center_Frequency>
SYSTem:PRESet:BLUEtooth:STANdard?
```

Arguments BR is the Basic Rate.
LE is the Low Energy Rate.
EDR is the Eye Diagram Rate.
<test_setup> ::= <string> the name of the test setup.
<Retaining_Current_Reflevel> ::= [1|0] where 1 indicates this value is checked in the view; 0 indicates the value is unchecked.
<Retaining_Center_Frequency> ::= [1|0] where 1 indicates this value is checked in the view; 0 indicates the value is unchecked.

Examples `SYSTEM:PRESET:BLUETOOTH:STANDARD` might return BR,MODD,0,1 indicating that the selected standard type is Basic rate, the test pattern is Modulation. The 1 indicates the Retaining Current Reflevel is checked; the 0 indicates the Retaining Center Frequency is unchecked.

SYSTem:PRESet:DPX (No Query Form)

Restores the analyzer to the defaults for the DPX preset type.

Conditions Measurement views: All

Group System commands

Syntax
```
SYSTem:PRESet:DPX { SWEPt | REALtime | ZERospan }
```

Arguments `SWEPt` restores the setup of the Swept DPX measurement.

`REALtime` restores the setup of the Realtime DPX measurement (110 MHz span).

`ZERospan` restores the setup of the Zero Span DPX measurement.

Examples `SYSTEM:PRESET:DPX REALtime` restores the DPX, Spectrum to realtime.

SYSTem:PRESet:DPX:ACTion

Sets or queries the preset action for the DPX preset type.

Conditions Measurement views: All

Group System commands

Syntax
```
SYSTem:PRESet:DPX:ACTion { RECall | SHOW }
SYSTem:PRESet:DPX:ACTion?
```

Arguments `RECall` recalls the selected preset for the DPX preset type.

`SHOW` lists the available presets for the DPX preset type.

Examples `SYSTEM:PRESET:DPX:ACTION SHOW` lists the available presets for the DPX preset type.

SYSTem:PRESet:DPX:SELected

Sets or queries the presets for the DPX preset type.

Conditions Measurement views: All

Group System commands

Syntax
```
SYSTem:PRESet:DPX:SELected { OPEn | SWEPt | REALtime |
ZERospan }
SYSTem:PRESet:DPX:SELected?
```

Arguments `OPEn` sets the DPX measurement to Open the DPX display.

`SWEPt` sets the DPX measurement to Swept mode.

`REALtime` sets the DPX measurement to Real-time mode (110 MHz span).

`ZERospan` sets the DPX measurement to Zero Span mode.

Examples `SYSTEM:PRESET:DPX:SELECTED REALtime` sets the DPX, Spectrum to realtime

SYSTem:PRESet:LTE:STANdard

Sets or queries the following: Preset, Test Setup, frame structure channel bandwidth, Base Station category (for LTE ACLR and SEM test setups), Adjacent channel type (for LTE ACLR test setup), Retain current Center Frequency setting, and Retain current Reference Level.

Conditions Measurement view: Standards Presets

Group System commands

Syntax

```
SYSTem:PRESet:LTE:STANdard { MODUlation | ACLR | CS | SEM,
FDD | TDD, BW1PT4M | BW3M | BW5M | BW10M | BW15M | BW20M,
WACA | WACB1 | WACB2 | HACAB | LACAB | MACAB, UTRA | EUTRA,
CR1PT28M | CR3PT84M | CR7PT68M, 0 | 1, 0 | 1 ]
SYSTem:PRESet:LTE:STANdard?
```

Arguments MODUlation | ACLR | CS | SEM, FDD | TDD, BW1PT4M | BW3M | BW5M | BW10M | BW15M | BW20M, WACA | WACB1 | WACB2 | HACAB | LACAB | MACAB, UTRA | EUTRA, CR1PT28M | CR3PT84M | CR7PT68M, 0 | 1, 0 | 1

Returns See Arguments.

Examples `SYSTEM:PRESET:LTE:STANDARD?` might return MOD,FDD,BW1PT4M,0,1, indicating that the test setup is Cell ID, Frame Structure, Channel BW, Retain Center Frequency, Retain Reference Level.

SYSTem:PRESet[:MAIN] (No Query Form)

Restores the analyzer to the defaults. This command is equivalent to the SYSTem:PRESet command.

Conditions Measurement views: All

Group System commands

Syntax

```
SYSTem:PRESet[:MAIN] { MAIN|}
SYSTem:PRESet[:MAIN]?
```

Arguments `CURRent` restores the analyzer to the current version of the main preset.

`V1` restores the analyzer to the original version of the main preset.

`V2` restores the analyzer to the Full Spectrum version of the main preset.

Examples `SYSTEM:PRESET[:MAIN] CURRENT` restores the analyzer to the defaults of the current version.

SYSTem:PRESet:MAIN:ACTion

Sets or queries the preset action for the main preset type.

Conditions Measurement views: All

Group System commands

Syntax
```
SYSTem:PRESet:MAIN:ACTion { RECall | SHOW }
SYSTem:PRESet:MAIN:ACTion?
```

Arguments `RECall` recalls the selected preset for the main preset type.

`SHOW` lists the available presets for the main preset type.

Examples `SYSTEM:PRESET:MAIN:ACTION SHOW` lists the available presets for the main preset type.

SYSTem:PRESet:MAIN:SELected

Sets or queries the presets for the Main preset type.

Conditions Measurement views: All

Group System commands

Syntax
```
SYSTem:PRESet:MAIN:SELected { CURRent | V1 | V2 }
SYSTem:PRESet:MAIN:SELected?
```

Arguments `CURRent` selects the current version of the Main presets.

`V1` selects the previous version of the main preset.

`V2` selects Full Spectrum Sweep version of the current main preset.

Examples `SYSTEM:PRESET:MAIN:SELECTED CURRent` selects the current version of the Main presets.

SYSTem:PRESet:P25:STANdard (No Query Form)

This command restores the analyzer defaults for the P25 Standards preset.

Conditions Measurement view: Any P25 measurement

Group System commands

Syntax `SYSTem:PRESet:P25:STANdard <{ PHASe1 | PHASe2 }>,<{ C4FM | HCPM | HDQPSK }>,<{ 1 | 0 }>`

Arguments PHASe1 and PHASe2 are the P25 standard types.

C4FM, HCPM, HDQPSK are the P25 modulation types.

1 = retain center frequency; 0 = do not retain center frequency.

Returns <standard> is the P25 standard type: Phase1, Phase2.

<modulation type> is the P25 modulation type: C4FM, HCPM (Inbound), HDQPSK (Outbound).

<{ 1 | 0 }> controls whether or not the center frequency is retained: 0 = not retained; 1 = retained.

Examples `SYSTEM:PRESET:P25:STANDARD?` might return `PHASE1,C4FM,0`, indicating that the P25 preset is set to standard type Phase 1, modulation type C4FM, and that the center frequency is set to not be retained.

SYSTem:PRESet:STANdards (No Query Form)

Restores the analyzer to the defaults for the WLAN Standards preset type. To restore the analyzer to a preset file, use the command `SYSTem:PRESet:USER.`

Conditions Measurement views: All

Group System commands

Syntax `SYSTem:PRESet:STANdards <WLAN>`

Arguments `<WLAN>` specifies to restore the analyzer to the WLAN standards preset type.

Examples `SYSTEM:PRESET:STANDARDS "WLAN"` restores the analyzer to the defaults for the WLAN Standards preset type.

SYSTem:PRESet:STANdards:ACTion

Sets or queries the preset action for the Standards preset type.

Conditions Measurement views: All

Group System commands

Syntax
```
SYSTem:PRESet:STANdards:ACTion { RECall | SHOW }
SYSTem:PRESet:STANdards:ACTion?
```

Arguments `RECall` recalls the selected preset for the Standards preset type.

`SHOW` lists the available presets for the Standards preset type.

Examples `SYSTEM:PRESET:STANDARDS:ACTION SHOW` lists the available presets for the Standards preset type.

SYSTem:PRESet:USER (No Query Form)

Restores the analyzer to the defaults for the user preset type.

Conditions Measurement views: All

Group System commands

Syntax `SYSTem:PRESet:USER <file_name>`

Arguments `<file_name> :: <string>` specifies the user presets file to restore. The file extension is .Setup. You can omit the extension.

Examples `SYSTEM:PRESET:USER "Spectrum"` restores the defaults of the file "Spectrum.Setup" from the C:\RSA5100B Files\User Presets directory.

SYSTem:PRESet:USER:ACTion

Sets or queries the preset action for the user preset type.

Conditions Measurement views: All

Group System commands

Syntax
```
SYSTem:PRESet:USER:ACTion { RECall | SHOW }
SYSTem:PRESet:USER:ACTion?
```

Arguments `RECall` recalls the selected preset for the user preset type.

`SHOW` lists the available presets for the user preset type.

Examples `SYSTEM:PRESET:USER:ACTION SHOW` lists the available presets for the User preset type.

SYSTem:PRESet:USER:SELected

Sets or queries the presets for the User preset type.

Conditions Measurement views: All

Group System commands

Syntax
```
SYSTem:PRESet:USER:SELected <file_name>
SYSTem:PRESet:USER:SELected?
```

Arguments `<file_name> :: <string>` specifies the User presets file to set. The file extension is .Setup. You can omit the extension.

Examples `SYSTEM:PRESET:USER:SELECTED "Spectrum"` sets the User preset to "Spectrum.Setup" saved in C:\RSA5100B Files\User Presets directory.

SYSTem:PRESet:WLAN:STANdard

Sets or queries the standard and bandwidth presets for the WLAN standards preset type. The first enumeration is the 802.11 standard; the second enumeration is the bandwidth.

Conditions Measurement views: All

Group System commands

Syntax

```
SYSTem:PRESet:WLAN:STANdard { B | AC,[BW20|BW40|BW80|BW160]
| [A,G],BW20 | P,[BW5|BW10|BW20] | J,[BW10|BW20] |
N,[BW20|BW40] }
SYSTem:PRESet:WLAN:STANdard?
```

Arguments `B` selects the 802.11b WLAN standard preset options.

`AC,[BW20|BW40|BW80|BW160]` selects the 802.11b WLAN standard preset options.

`[A,G],BW20` selects the 802.11a or g WLAN standard preset options.

`P,[BW5|BW10|BW20]` selects the 802.11p WLAN standard preset options.

`J,[BW10|BW20]` selects the 802.11j WLAN preset options.

`N,[BW20|BW40]` selects the 802.11n WLAN standard preset options.

Examples `SYSTEM:PRESET:WLAN:STANDARD P, BW5` selects 802.11p as the WLAN standard to use and 5 MHz as the bandwidth preset.

SYSTem:TIME

Sets or queries the time (hours, minutes, and seconds). This command is equivalent to the time setting through the Windows Control Panel.

Conditions Measurement views: All

Group System commands

Syntax
```
SYSTem:TIME <hour>,<minute>,<second>
SYSTem:TIME?
```

Related Commands SYSTem:DATE

Arguments <hour> :: <NRf> specifies the hours. Range: 0 to 23.
<minute> :: <NRf> specifies the minutes. Range: 0 to 59.
<second> :: <NRf> specifies the seconds. Range: 0 to 59.
These values are rounded to the nearest integer.

*RST has no effect on the settings.

Examples SYSTEM:TIME10,15,30 sets the time to 1015:30.

SYSTem:VERSion? (Query Only)

Returns the SCPI version number for which the analyzer complies.

Conditions Measurement views: All

Group System commands

Syntax
```
SYSTem:VERSion?
```

Arguments None

Returns <NR2> has the form YYYY.V where the Ys represent the year-version (for example, 1999) and the V represents an approved revision number for that year.

Examples SYSTEM:VERSION? might return 1999.0 for the SCPI version.

TRACe:{AM|FM|PM}:DETection

Sets or queries the display detector, the method to be used for decimating traces to fit the available horizontal space on screen. The number of horizontal pixels on screen is generally smaller than that of waveform data points. When actually

displayed, the waveform data is therefore thinned out, according to the number of pixels, for being compressed.

Conditions Measurement views: AM, FM, PM

Group Trace commands

Syntax
```
TRACe:{AM|FM|PM}:DETection { AVERage | POSitive | NEGative }
TRACe:{AM|FM|PM}:DETection?
```

Arguments `AVERage` displays the average data value for each pixel.

`POSitive` displays the maximum data value for each pixel.

`NEGative` displays the minimum data value for each pixel.

Examples `TRACE:AM:DETECTION AVERage` specifies that the trace displays the average data value for each pixel.

TRACe:{AM|FM|PM}:FREeze

Determines whether or not to freeze the trace display in the AM/FM/PM view.

Conditions Measurement views: AM, FM, PM

Group Trace commands

Syntax
```
TRACe:{AM|FM|PM}:FREeze { OFF | ON | 0 | 1 }
TRACe:{AM|FM|PM}:FREeze?
```

Arguments `OFF` or `0` updates the trace display normally.

`ON` or `1` stops updating the trace display.

Examples `TRACE:AM:FREEZE ON` freezes the trace display.

TRACe:{AM|FM|PM}:FUNCtion

Sets or queries the trace function in the AM/FM/PM view.

Conditions Measurement views: AM, FM, PM

Group Trace commands

Syntax
```
TRACe:{AM|FM|PM}:FUNCtion { NORMal }
TRACe:{AM|FM|PM}:FUNCtion?
```

Arguments `NORMal` selects the normal display.

Examples `TRACE:AM:FUNCTION NORMal` selects the normal display in the AM view.

TRACe:DIQVtime:ENABle:I

Determines whether to show or hide the I trace in the Demod I&Q versus Time measurement.

Conditions Measurement views: Demod I&Q versus Time

Group Trace commands

Syntax
```
TRACe:DIQVtime:ENABle:I { OFF | ON | 0 | 1 }
TRACe:DIQVtime:ENABle:I?
```

Arguments `OFF` or `0` hides the I trace.

`ON` or `1` shows the I trace.

Examples `TRACE:DIQVTIME:ENABLE:ION` shows the I trace in the Demod I&Q versus Time measurement.

TRACe:DIQVtime:ENABle:Q

Determines whether to show or hide the Q trace in the Demod I&Q versus Time measurement.

Conditions Measurement views: Demod I&Q versus Time

Group Trace commands

Syntax

```
TRACe:DIQVtime:ENABle:Q { OFF | ON | 0 | 1 }
TRACe:DIQVtime:ENABle:Q?
```

Arguments

OFF or 0 hides the Q trace.

ON or 1 shows the Q trace.

Examples

TRACE:DIQVTIME:ENABLE:QON shows the Q trace in the Demod I&Q versus Time measurement.

TRACe:DIQVtime:MODE

Sets or queries whether to display the Demod I&Q vs Time trace as vectors or symbols (points).

Conditions

Measurement view: Demod I&Q vs Time

Group

Trace commands

Syntax

```
TRACe:DIQVtime:MODE { VECTors | SYMBols }
```

Arguments

VECTors displays the trace in a format that uses lines to connecting points in the trace display.

SYMBols displays the trace in a format that does not use lines to connect points in the trace display.

Examples

TRACE:DIQVTIME:MODE SYMBOLS sets the display to show symbols without lines connecting points.

TRACe:DIQVtime:SELect:I

Selects the I trace in the Demod I&Q versus Time. The query version of this command returns whether the I trace is selected or not.

Conditions

Measurement views: Demod I&Q versus Time

Group

Trace commands

Syntax

```
TRACe:DIQVtime:SELect:I
TRACe:DIQVtime:SELect:I?
```

Arguments None

Returns `{ 0 | 1 }`

`0` indicates that the I trace is deselected.

`1` indicates that the I trace is selected.

Examples `TRACE:DIQVTIME:SELECT:I` selects the I trace in the Demod I&Q versus Time.

TRACe:DIQVtime:SELect:Q

Selects the Q trace in the Demod I&Q versus Time. The query version of this command returns whether the Q trace is selected or not.

Conditions Measurement views: Demod I&Q versus Time

Group Trace commands

Syntax

```
TRACe:DIQVtime:SELect:Q
TRACe:DIQVtime:SELect:Q?
```

Arguments None

Returns `{ 0 | 1 }`

`0` indicates that the Q trace is deselected.

`1` indicates that the Q trace is selected.

Examples `TRACE:DIQVTIME:SELECT:Q` selects the Q trace in the Demod I&Q versus Time.

TRACe:EDIagram:ENABle:I

Determines whether to show or hide the I trace in the eye diagram.

Conditions Measurement views: Eye diagram

Group Trace commands

Syntax
```
TRACe:EDIagram:ENABle:I { OFF | ON | 0 | 1 }
TRACe:EDIagram:ENABle:I?
```

Arguments `OFF` or `0` hides the I trace.

`ON` or `1` shows the I trace.

Examples `TRACE:EDIAGRAM:ENABLE:ION` shows the I trace in the eye diagram.

TRACe:EDIagram:ENABle:Q

Determines whether to show or hide the Q trace in the eye diagram.

Conditions Measurement views: Eye diagram

Group Trace commands

Syntax
```
TRACe:EDIagram:ENABle:Q { OFF | ON | 0 | 1 }
TRACe:EDIagram:ENABle:Q?
```

Arguments `OFF` or `0` hides the Q trace.

`ON` or `1` shows the Q trace.

Examples `TRACE:EDIAGRAM:ENABLE:QON` shows the Q trace in the eye diagram.

TRACe:EDIagram:SELect:I

Selects the I trace in the eye diagram. The query version of this command returns whether the I trace is selected or not.

Conditions Measurement views: Eye diagram

Group Trace commands

Syntax

```
TRACe:EDIagram:SELect:I
TRACe:EDIagram:SELect:I?
```

Arguments None

Returns

```
{ 0 | 1 }
```

0 indicates that the I trace is deselected.

1 indicates that the I trace is selected.

Examples `TRACE:EDIAGRAM:SELECT:I` selects the I trace in the eye diagram.

TRACe:EDIagram:SELect:Q

Selects the Q trace in the eye diagram. The query version of this command returns whether the Q trace is selected or not.

Conditions Measurement views: Eye diagram

Group Trace commands

Syntax

```
TRACe:EDIagram:SELect:Q
TRACe:EDIagram:SELect:Q?
```

Arguments None

Returns

```
{ 0 | 1 }
```

0 indicates that the Q trace is deselected.

1 indicates that the Q trace is selected.

Examples `TRACE:EDIAGRAM:SELECT:Q` selects the Q trace in the eye diagram.

TRACe:EVM:MODE

Sets or queries whether to display the EVM vs Time trace as vectors or symbols (points).

Conditions Measurement view: EVM vs Time

Group Trace commands

Syntax `TRACe:EVM:MODE { VECTors | SYMBols }`

Arguments VECTors displays the trace in a format that uses lines to connecting points in the trace display.

SYMBols displays the trace in a format that does not use lines to connect points in the trace display.

Examples `TRACE:EVM:MODE VECTORS` sets the display to connect points in the display with lines.

TRACe:FDVTime:MODE

Sets or queries whether to display the Frequency Deviation vs Time trace as vectors or symbols (points).

Conditions Measurement view: Freq Dev vs Time

Group Trace commands

Syntax `TRACe:FDVTime:MODE { VECTors | SYMBols }`

Arguments VECTors displays the trace in a format that uses lines to connecting points in the trace display.

SYMBols displays the trace in a format that does not use lines to connect points in the trace display.

Examples `TRACE:FDVTIME:MODE VECTORS` sets the display to connect points in the display with lines.

TRACe:{FSETtling|PSETtling}:AVERage:COUNt

Sets or queries the number of traces to combine. This command is effective when you select AVERage with the TRACe:{FSETtling|PSETtling}:AVERage:ENABle command.

Conditions Measurement views: Frequency and Phase Settling Time

Group Trace commands

Syntax

```
TRACe:{FSETtling|PSETtling}:AVERage:COUNt <number>
TRACe:{FSETtling|PSETtling}:AVERage:COUNt?
```

Related Commands TRACe:{FSETtling|PSETtling}:AVERage:ENABle

Arguments <number> :: <NR1> specifies the number of traces to combine for averaging. Range: 1 to 10000.

Examples TRACE1:PSETTLING:AVERAGE:COUNT 64 sets the average count to 64 for Trace 1.

TRACe:{FSETtling|PSETtling}:AVERage:COUNt:CURRent? (Query Only)

Queries the current running average count value. The returned value is valid (non-zero) only while the instrument is acquiring and averaging a signal. Use the TRACe:{FSETtling|PSETtling}:AVERage:COUNt command to determine the setting for Average.

Conditions Measurement views: Frequency and Phase Settling Time

Group Trace commands

Syntax

```
TRACe:{FSETtling|PSETtling}:AVERage:COUNt:CURRent?
```

Related Commands TRACe:{FSETtling|PSETtling}:AVERage:COUNt

Examples TRACE1:PSETTLING:AVERAGE:COUNT? might return the value 64.

TRACe:{FSETtling|PSETtling}:AVERage:ENABle

Determines whether to enable or disable averaging the trace(s).

Conditions Measurement views: Frequency and Phase Settling Time

Group Trace commands

Syntax
```
TRACe:{FSETtling|PSETtling}:AVERage:ENABle { OFF | ON | 0 | 1 }
TRACe:{FSETtling|PSETtling}:AVERage:ENABle?
```

Arguments `OFF` or `0` disables averaging.

`ON` or `1` enables averaging.

Examples `TRACE:PSETTLING:AVERAGE:ENABLE ON` enables the Phase Settling Time averaging.

TRACe:{FSETtling|PSETtling}:RESet (No Query Form)

If trace averaging is enabled, this command resets the current averaged trace to 0.

Conditions Measurement views: Frequency and Phase Settling Time

Group Trace commands

Syntax
```
TRACe:{FSETtling|PSETtling}:RESet
```

Examples `TRACE1:PSETTLING:RESET` resets the current averaged trace to 0.

TRACe:{FSETtling|PSETtling}:SMOothing:COUNt

Sets or queries the number of data points to take the moving average for smoothing the traces. This command applies to both Trace 1 and Trace 2.

Conditions Measurement views: Frequency and Phase Settling Time

Group Trace commands

Syntax
```
TRACe:{FSETtling|PSETtling}:SMOothing:COUNt <number>
TRACe:{FSETtling|PSETtling}:SMOothing:COUNt?
```

Arguments <number> :: <NR1> specifies the number of data points to take the moving average for smoothing. Range: 2 to 1000.

Examples TRACE:FSETTLING:SMOOTHING:COUNT 16 sets the Frequency Settling Time smoothing count to 16.

TRACe:{FSETtling|PSETtling}:SMOothing:ENABle

Determines whether to enable or disable smoothing the trace(s).

Conditions Measurement views: Frequency and Phase Settling Time

Group Trace commands

Syntax
```
TRACe:{FSETtling|PSETtling}:SMOothing:ENABle { OFF | ON |
0 | 1 }
TRACe:{FSETtling|PSETtling}:SMOothing:ENABle?
```

Arguments OFF or 0 disables smoothing.

ON or 1 enables smoothing.

Examples TRACE:PSETTLING:SMOOTHING:ENABLE ON enables the Phase Settling Time smoothing.

TRACe:FVTime

Determines whether or not to show the trace in the Frequency versus Time view.

Conditions Measurement views: Frequency versus Time

Group Trace commands

Syntax

```
TRACe:FVTime { OFF | ON | 0 | 1 }
TRACe:FVTime?
```

Arguments

OFF or 0 hides the trace in the Frequency versus Time view.

ON or 1 shows the trace in the Frequency versus Time view.

Examples

TRACE:FVTIMEON shows the trace in the Frequency versus Time view.

TRACe:FVTime:AVERage:COUNt

Sets or queries the number of traces to combine. This command is effective when you select AVERage with the TRACe:FVTime:FUNCtion command.

Conditions

Measurement views: Frequency versus Time

Group

Trace commands

Syntax

```
TRACe:FVTime:AVERage:COUNt <number>
TRACe:FVTime:AVERage:COUNt?
```

Arguments

<number> :: <NR1> specifies the number of traces to combine for averaging. Range: 1 to 10000.

Examples

TRACE:FVTIME:AVERAGE:COUNT64 sets the average count to 64.

TRACe:FVTime:COUNt

Sets or queries how many acquisitions run in the single acquisition mode for the Max or Min Hold trace in the Frequency versus Time measurement. This command is effective when TRACe:FVTime:FUNCtion is set to MAXHold or MINHold and INITiate:CONTinuous is set to OFF.

Conditions

Measurement views: Frequency versus Time

Group

Trace commands

Syntax

```
TRACe:FVTime:COUNt <number>
TRACe:FVTime:COUNt?
```

Arguments <number> :: <NR1> specifies the count for Max/Min Hold. Range: 1 to 10000.

Examples TRACe:FVTime:COUNt 32 sets the count to 32 for the Max/Min Hold trace.

TRACe:FVTime:COUNt:ENABle

Determines whether to enable or disable the count for the Max or Min Hold trace in the Frequency versus Time view. This command is effective when TRACe:FVTime:FUNCtion is set to MAXHold or MINHold.

Conditions Measurement views: Frequency versus Time

Group Trace commands

Syntax
```
TRACe:FVTime:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe:FVTime:COUNt:ENABle?
```

Arguments OFF or 0 disables the count for the Max/Min Hold trace.

ON or 1 enables the count for the Max/Min Hold trace.

Examples TRACe:FVTime:COUNt:ENABle ON enables the Max/Min Hold count.

TRACe:FVTime:COUNt:RESet (No Query Form)

Clears the Max or Min Hold data and counter, and restarts the process in the Frequency versus Time view. This command is effective when TRACe:FVTime:FUNCtion is set to MAXHold or MINHold.

Conditions Measurement views: Frequency versus Time

Group Trace commands

Syntax TRACe:FVTime:COUNt:RESet

Arguments None

Examples `TRACe:FVTime:COUNt:RESet` clears the Max/Min Hold data and counter, and restarts the process.

TRACe:FVTime:FREeze

Determines whether or not to freeze the display of the trace in the Frequency versus Time measurement.

Conditions Measurement views: Frequency versus Time

Group Trace commands

Syntax
```
TRACe:FVTime:FREeze { OFF | ON | 0 | 1 }
TRACe:FVTime:FREeze?
```

Arguments `OFF` or `0` updates the display of the trace normally.

`ON` or `1` stops updating the display of the trace.

Examples `TRACE:FVTIME:FREEZEON` stops updating the display of the trace.

TRACe:FVTime:FUNCtion

Sets or queries the trace function in the Frequency versus Time measurement.

Conditions Measurement views: Frequency versus Time

Group Trace commands

Syntax
```
TRACe:FVTime:FUNCtion { NORMal | AVERage | MAXHold | MINHold }
TRACe:FVTime:FUNCtion?
```

Arguments `NORMal` selects the normal waveform display.

`AVERage` selects the Average display that indicates the average frequency drift at each time point.

`MAXHold` selects the Max Hold display that indicates the maximum frequency drift at each time point.

MINHold selects the Min Hold display that indicates the minimum frequency drift at each time point.

Examples TRACE:FVTIME:FUNCTIONMAXHold displays the Max Hold trace in the Frequency versus Time measurement.

TRACe:IQVTime:AVERage:COUNt

Sets or queries the number of traces to combine. This command works for both I and Q traces when you select AVERage in the TRACe:IQVTime:FUNCtion command.

Conditions Measurement views: RF I&Q versus Time

Group Trace commands

Syntax
```
TRACe:IQVTime:AVERage:COUNt <number>
TRACe:IQVTime:AVERage:COUNt?
```

Arguments <number> :: <NR1> specifies the number of traces to combine for averaging. Range: 1 to 10000.

Examples TRACE:IQVTIME:AVERAGE:COUNT64 sets the average count to 64.

TRACe:IQVTime:COUNt

Sets or queries how many acquisitions run in the single acquisition mode for the Max or Min Hold trace in the RF I&Q versus Time measurement. This command is effective when TRACe:IQVTime:FUNCtion is set to MAXHold or MINHold and INITiate:CONTinuous is set to OFF.

Conditions Measurement views: RF I&Q versus Time

Group Trace commands

Syntax
```
TRACe:IQVTime:COUNt <number>
TRACe:IQVTime:COUNt?
```

Arguments `<number> ::  <NR1>` specifies the count for Max/Min Hold.
Range: 1 to 10000.

Examples `TRACE:IQVTIME:COUNT32` sets the count to 32 for the Max/Min Hold trace.

TRACe:IQVTime:COUNt:ENABle

Determines whether to enable or disable the count for the Max or Min Hold trace in the RF I&Q versus Time view. This command is effective when TRACe:IQVTime:FUNCtion is set to MAXHold or MINHold.

Conditions Measurement views: RF I&Q versus Time

Group Trace commands

Syntax
```
TRACe:IQVTime:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe:IQVTime:COUNt:ENABle?
```

Arguments `OFF` or `0` disables the count for the Max/Min Hold trace.

`ON` or `1` enables the count for the Max/Min Hold trace.

Examples `TRACE:IQVTIME:COUNT:ENABLEON` enables the Max/Min Hold count.

TRACe:IQVTime:COUNt:RESet (No Query Form)

Clears the Max or Min Hold data and counter, and restarts the process in the RF I&Q versus Time view. This command is effective when TRACe:IQVTime:FUNCtion is set to MAXHold or MINHold.

Conditions Measurement views: RF I&Q versus Time

Group Trace commands

Syntax `TRACe:IQVTime:COUNt:RESet`

Arguments None

Examples `TRACE:IQVTIME:COUNT:RESET` clears the Max/Min Hold data and counter, and restarts the process.

TRACe:IQVTime:DETection

Sets or queries the detection method for the RF I&Q versus Time view.

Conditions Measurement views: RF I&Q versus Time

Group Trace commands

Syntax

```
TRACe:IQVTime:DETection { AVERage | POSitive | NEGative |
POSNegative | SAMPle }
TRACe:IQVTime:DETection?
```

Arguments `AVERage` displays the average data value for each pixel.

`POSitive` displays the maximum data value for each pixel.

`NEGative` displays the minimum data value for each pixel.

`POSNegative` displays the maximum and minimum data values for each pixel.

`SAMPle` displays the first sample value received for each pixel.

Examples `TRACE:IQVTIME:DETECTIONAVERAGE` enables display of the average value for each pixel.

TRACe:IQVTime:ENABle:I

Determines whether to show or hide the I trace in the RF I&Q versus Time measurement.

Conditions Measurement views: RF I&Q versus Time

Group Trace commands

Syntax

```
TRACe:IQVTime:ENABle:I { OFF | ON | 0 | 1 }
TRACe:IQVTime:ENABle:I?
```

Arguments `OFF` or `0` hides the I trace.

`ON` or `1` shows the I trace.

Examples `TRACE:IQVTIME:ENABLE:ION` shows the I trace in the RF I&Q versus Time measurement.

TRACe:IQVTime:ENABle:Q

Determines whether to show or hide the Q trace in the RF I&Q versus Time measurement.

Conditions Measurement views: RF I&Q versus Time

Group Trace commands

Syntax
```
TRACe:IQVTime:ENABle:Q { OFF | ON | 0 | 1 }
TRACe:IQVTime:ENABle:Q?
```

Arguments `OFF` or `0` hides the Q trace.

`ON` or `1` shows the Q trace.

Examples `TRACE:IQVTIME:ENABLE:QON` shows the Q trace in the IQ level versus Time measurement.

TRACe:IQVTime:FREeze

Determines whether to freeze the IQ traces in the RF I&Q versus Time measurement.

Conditions Measurement views: RF I&Q versus Time

Group Trace commands

Syntax
```
TRACe:IQVTime:FREeze { OFF | ON | 0 | 1 }
TRACe:IQVTime:FREeze?
```

Arguments `OFF` or `0` updates IQ trace display normally.

`ON` or `1` stops updating IQ trace display.

Examples `TRACE:IQVTIME:FREEZEON` freezes the IQ traces.

TRACe:IQVTime:FUNCtion

Sets or queries the trace function in the RF I&Q versus Time measurement.

Conditions Measurement views: RF I&Q versus Time

Group Trace commands

Syntax
```
TRACe:IQVTime:FUNCtion { NORMal | AVERage | MAXHold |
MINHold }
TRACe:IQVTime:FUNCtion?
```

Arguments `NORMal` selects the normal waveform display.

`AVERage` selects the Average display that indicates the average signal level at each time point.

`MAXHold` selects the Max Hold display that indicates the maximum signal level at each time point.

`MINHold` selects the Min Hold display that indicates the minimum signal level at each time point.

Examples `TRACE:IQVTIME:FUNCTIONMAXHold` displays the Max Hold trace in the IQ level versus Time measurement.

TRACe:IQVTime:SELect:I

Determines whether or not to select the I trace to obtain the maximum and minimum measurement results.

Conditions Measurement views: RF I&Q versus Time

Group Trace commands

Syntax

```
TRACe:IQVTime:SELect:I { OFF | ON | 0 | 1 }
TRACe:IQVTime:SELect:I?
```

Related Commands TRACe:IQVTime:SELect:Q

Arguments `OFF` or `0` deselects the I trace.

`ON` or `1` selects the I trace.

Executing TRACe:IQVTime:SELect:I ON sets TRACe:IQVTime:SELect:Q OFF.

Examples `TRACE:IQVTIME:SELECT:ION` selects the I trace in the RF I&Q versus Time measurement.

TRACe:IQVTime:SELect:Q

Determines whether or not to select the Q trace to obtain the maximum and minimum measurement results.

Conditions Measurement views: RF I&Q versus Time

Group Trace commands

Syntax

```
TRACe:IQVTime:SELect:Q { OFF | ON | 0 | 1 }
TRACe:IQVTime:SELect:Q?
```

Related Commands TRACe:IQVTime:SELect:I

Arguments `OFF` or `0` deselects the Q trace.

`ON` or `1` selects the Q trace.

Executing TRACe:IQVTime:SELect:Q ON sets TRACe:IQVTime:SELect:I OFF.

Examples `TRACE:IQVTIME:SELECT:QON` selects the Q trace in the RF I&Q versus Time measurement.

TRACe:MERRor:MODE

Sets or queries whether to display the Magnitude error trace as vectors or symbols (points).

Conditions Measurement view: Magnitude error

Group Trace commands

Syntax `TRACe:MERRor:MODE { VECTors | SYMBols }`

Arguments VECTors displays the trace in a format that uses lines to connecting points in the trace display.

SYMBols displays the trace in a format that does not use lines to connect points in the trace display.

Examples `TRACE:MERRor:MODE SYMBOLS` sets the display to show symbols without lines connecting points.

TRACe:OBW:MAXHold

Determines whether or not to perform a Max Hold on the spectrum data for the Occupied Bandwidth trace.

Conditions Measurement views: Occupied Bandwidth

Group Trace commands

Syntax
```
TRACe:OBW:MAXHold { OFF | ON | 0 | 1 }
TRACe:OBW:MAXHold?
```

Arguments `OFF` or `0` does not perform a Max Hold on the spectrum data.

`ON` or `1` performs a Max Hold on the spectrum data.

Examples `TRACE:OBW:MAXHOLDON` performs a Max Hold on the spectrum data for the Occupied Bandwidth trace.

TRACe:OFDM:CONSte

Determines whether or not to show the trace in the OFDM Constellation view.

Conditions Measurement views: OFDM

Group Trace commands

Syntax
```
TRACe:OFDM:CONSte { OFF | ON | 0 | 1 }
TRACe:OFDM:CONSte?
```

Arguments OFF or 0 hides the trace in the view.

ON or 1 shows the trace in the view.

Examples TRACE:OFDM:CONSTE ON shows the trace in the OFDM Constellation view

TRACe:OFDM:CONSte:FREeze

Determines whether or not to freeze the display of the trace in the OFDM Constellation view.

Conditions Measurement views: OFDM

Group Trace commands

Syntax
```
TRACe:OFDM:CONSte:FREeze { OFF | ON | 0 | 1 }
TRACe:OFDM:CONSte:FREeze?
```

Arguments OFF or 0 updates the display of the trace normally.

ON or 1 stops updating the display of the trace.

Examples TRACE:OFDM:CONSTE:FREEZE ON freezes the trace in the OFDM Constellation view.

TRACe:OFDM:FLATness:FREeze

Sets or queries whether to freeze the average trace on the OFDM Spectral Flatness display. When the freeze state is turned on, the trace stops updating.

Conditions Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group Trace commands

Syntax
```
TRACe:OFDM:FLATness:FREeze {0|1|OFF|ON}
TRACe:OFDM:FLATness:FREeze?
```

Arguments ON or 1 turns on the freeze state, causing the trace to stop updating.

OFF or 0 turns off the freeze state.

Returns Outputs the frozen status of the selected trace as an integral value. 1 indicates a frozen trace and 0 a non-frozen trace.

Examples TRAC:OFDM:FLAT:FRE 1 freezes the trace, causing it to stop updating.

TRACe:OFDM:FLATness:SHOW

Displays or hides the trace on the OFDM Spectral Flatness display.

Conditions Measurement view: OFDM

This command requires Option 22, “OFDM Measurements”.

Group Trace commands

Syntax
```
TRACe:OFDM:FLATness:SHOW {0|1|OFF|ON}
```

Arguments ON or 1 displays the trace.

OFF or 0 hides the trace.

Examples TRACE:OFDM:FLAT:SHOW 1 turns on the display of the trace.

TRACe:PERRor:MODE

Sets or queries whether to display the Frequency Deviation vs Time trace as vectors or symbols (points).

Conditions Measurement view: Phase error

Group Trace commands

Syntax `TRACe:PERRor:MODE { VECTors | SYMBols }`

Arguments VECTors displays the trace in a format that uses lines to connecting points in the trace display.

SYMBols displays the trace in a format that does not use lines to connect points in the trace display.

Examples `TRACE:PERROR:MODE VECTORS` sets the display to connect points in the display with lines.

TRACe:PHVTime

Determines whether or not to show the trace in the Phase versus Time view.

Conditions Measurement views: Phase versus Time

Group Trace commands

Syntax
```
TRACe:PHVTime { OFF | ON | 0 | 1 }
TRACe:PHVTime?
```

Arguments `OFF` or `0` hides the trace in the Phase versus Time view.

`ON` or `1` shows the trace in the Phase versus Time view.

Examples `TRACE:PHVTIMEON` shows the trace in the Phase versus Time view.

TRACe:PHVTime:AVERage:COUNt

Sets or queries the number of traces to combine. This command is effective when you select AVERage with the TRACe:PHVTime:FUNCtion command.

Conditions Measurement views: Phase versus Time

Group Trace commands

Syntax

```
TRACe:PHVTime:AVERage:COUNt <number>
TRACe:PHVTime:AVERage:COUNt?
```

Arguments

<number> :: <NR1> specifies the number of traces to combine for averaging. Range: 1 to 10000.

Examples

TRACE:PHVTIME:AVERAGE:COUNT64 sets the average count to 64.

TRACe:PHVTime:COUNt

Sets or queries how many acquisitions run in the single acquisition mode for the Max or Min Hold trace in the Phase versus Time measurement. This command is effective when TRACe:PHVTime:FUNCtion is set to MAXHold or MINHold and INITiate:CONTinuous is set to OFF.

Conditions Measurement views: Phase versus Time

Group Trace commands

Syntax

```
TRACe:PHVTime:COUNt <number>
TRACe:PHVTime:COUNt?
```

Arguments <number> :: <NR1> specifies the count for Max/Min Hold.
Range: 1 to 10000.

Examples TRACE:PHVTIME:COUNT32 sets the count to 32 for the Max/Min Hold trace.

TRACe:PHVTime:COUNt:ENABle

Determines whether to enable or disable the count for the Max or Min Hold trace in the Phase versus Time view. This command is effective when TRACe:PHVTime:FUNCtion is set to MAXHold or MINHold.

Conditions Measurement views: Phase versus Time

Group Trace commands

Syntax

```
TRACe:PHVTime:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe:PHVTime:COUNt:ENABle?
```

Related Commands

Arguments OFF or 0 disables the count for the Max/Min Hold trace.

ON or 1 enables the count for the Max/Min Hold trace.

Examples TRACE:PHVTIME:COUNT:ENABLEON enables the Max/Min Hold count.

TRACe:PHVTime:COUNt:RESet (No Query Form)

Clears the Max or Min Hold data and counter, and restarts the process in the Phase versus Time view. This command is effective when TRACe:PHVTime:FUNCtion is set to MAXHold or MINHold.

Conditions Measurement views: Phase versus Time

Group Trace commands

Syntax
```
TRACe:PHVTime:COUNt:RESet
```

Arguments None

Examples `TRACE:PHVTIME:COUNT:RESET` clears the Max/Min Hold data and counter, and restarts the process.

TRACe:PHVTime:FREeze

Determines whether to freeze the trace display in the Phase versus Time measurement.

Conditions Measurement views: Phase versus Time

Group Trace commands

Syntax
```
TRACe:PHVTime:FREeze { OFF | ON | 0 | 1 }
TRACe:PHVTime:FREeze?
```

Arguments `OFF` or `0` updates the trace display normally.

`ON` or `1` stops updating trace display.

Examples `TRACE:PHVTIME:FREEZEON` stops updating trace display.

TRACe:PHVTime:FUNCtion

Sets or queries the trace function in the Phase versus Time measurement.

Conditions Measurement views: Phase versus Time

Group Trace commands

Syntax

```
TRACe:PHVTime:FUNCtion { NORMal | AVERage | MAXHold |
MINHold }
TRACe:PHVTime:FUNCtion?
```

Arguments

`NORMal` selects the normal waveform display.

`AVERage` selects the Average display that indicates the average phase drift at each time point.

`MAXHold` selects the Max Hold display that indicates the maximum phase drift at each time point.

`MINHold` selects the Min Hold display that indicates the minimum phase drift at each time point.

`POSNegative` displays the maximum and minimum data values for each pixel.

`SAMPle` displays the first sample value received for each pixel.

Examples `TRACE:PHVTIME:FUNCTIONMAXHold` displays the Max Hold trace in the Phase versus Time measurement.

TRACe:SEM:COUNt

Sets or queries how many acquisitions run in the single acquisition mode for multi-trace functions (Max Hold and Average) in the Spectral Emissions Mask measurement. This command is effective whenTRACe:SEM:FUNCtion is set to MAXHold or AVERage and INITiate:CONTinuous is set to OFF.

Conditions Measurement views: Spectral Emissions Mask

Group Trace commands

Syntax

```
TRACe:SEM:COUNt <number>
TRACe:SEM:COUNt?
```

Arguments `<number> ::  <NR1>` specifies the count for multi-trace functions. Range: 1 to 10000.

Examples `TRACE:SEM:COUNT 32` sets the count to 32 for multi-trace functions.

TRACe:SEM:COUNt:RESet (No Query Form)

Clears the multi-function (Max Hold or Average) data and counter and restarts the process in the Spectral Emissions Mask view. This command is effective when TRACe:SEM:FUNCtion is set to MAXHold or AVERage.

Conditions Measurement views: Spectral Emissions Mask

Group Trace commands

Syntax `TRACe:SEM:COUNt:RESet`

Arguments None

Examples `TRACE:SEM:COUNT:RESET` clears the multi-function data and counter, and restarts the process.

TRACe:SEM:FUNCtion

Sets or queries the trace function in the Spectral Emissions Mask view.

Conditions Measurement views: Spectral Emissions Mask

Group Trace commands

Syntax
```
TRACe:SEM:FUNCtion { NONE | MAXHold | AVERage }
TRACe:SEM:FUNCtion?
```

Arguments `NONE` selects normal display.

`MAXHold` selects the Max Hold display that indicates the maximum amplitude drift at each frequency point.

AVERage selects the Average display that indicates the average amplitude drift at each frequency point.

Examples TRACE:SEM:FUNCTION MAXHold displays the Max Hold trace in the Spectral Emissions Mask measurement.

TRACe:SGRam:DETection

Sets or queries the display detector (method to be used for decimating traces to fit the available horizontal space on screen). The number of horizontal pixels on screen is generally smaller than that of waveform data points. When actually displayed, the waveform data is therefore thinned out, according to the number of pixels, for being compressed.

Conditions Measurement views: Spectrogram

Group Trace commands

Syntax

```
TRACe:SGRam:DETection { AVERage | POSitive | NEGative |
CAVerage | CPEak | QUASipeak | AVGLog }
TRACe:SGRam:DETection?
```

Arguments

AVERage displays the average data value for each pixel.

POSitive displays the maximum data value for each pixel.

NEGative displays the minimum data value for each pixel.

CAVerage displays the CISPR average value for each pixel.

CPEak displays the CISPR peak value for each pixel.

QUASipeak displays the quasi-peak value for each pixel.

AVGLog displays the average data value of logs for each pixel.

Examples TRACE:SGRAM:DETECTIONPOSitive displays the maximum data value for each pixel.

TRACe:SGRam:FREeze

Determines whether or not to freeze the spectrogram display.

Conditions Measurement views: Spectrogram

Group Trace commands

Syntax

```
TRACe:SGRam:FREeze { OFF | ON | 0 | 1 }
TRACe:SGRam:FREeze?
```

Arguments `OFF` or `0` updates the display of the spectrogram normally.

`ON` or `1` stops updating the display of the spectrogram.

Examples `TRACE:SGRAM:FREEZEON` freezes the spectrogram display.

TRACe:SGRam:FUNCtion

NOTE. *This function is no longer available in software versions 2.3.0159 and later. Sending this command selects the correct [:SENSe]:SGRam:TIME[:SCALe]:MODe and :TRACe:SGRam:DETection to achieve the same result as with previous software versions. The query will return the equivalent trace function based on the MODe and DETection settings.*

Sets or queries the trace function for the specified trace in the spectrogram.

Conditions Measurement views: Spectrogram

Group Trace commands

Syntax

```
TRACe:SGRam:FUNCtion { NONE | AVERage | MAXHold | MINHold
| AVGLog }
TRACe:SGRam:FUNCtion?
```

Arguments `NONE` selects the normal spectrogram display.

`AVERage` selects the Average display that indicates the average signal level at each frequency point.

MAXHold selects the Max Hold display that indicates the maximum signal level at each frequency point.

MINHold selects the Min Hold display that indicates the minimum signal level at each frequency point.

AVGLog selects the Average of logs display that indicates the average signal level of logs at each frequency point.

Examples TRACE:SGRAM:FUNCTIONMAXHold selects the Max Hold display for the spectrogram.

TRACe:SGRam:FUNCtion:TIME

NOTE. *This command has a new function in software versions 2.3.159 and above.*

Sets the Spectrum Monitor mode values for Time per update. Sending this command sets [SENSe]:SGRam:TIME[:SCALe]:PER:UPDate:MINutes and [SENSe]:SGRam:WATerfall:DIRection based on the value sent in. The query form returns the number of minutes based on the new settings values.

Conditions Measurement views: Spectrogram

Group Trace commands

Syntax
```
TRACe:SGRam:FUNCtion:TIME <value>
TRACe:SGRam:FUNCtion:TIME?
```

Arguments <value> :: <NR1> specifies the time length in Spectrum Monitor mode. Range: 1 to 600 minutes and 60 seconds.

Examples TRACE:SGRAM:FUNCTION:TIME6E2 sets the Time/update value to 600 minutes.

TRACe:SGRam:SELect:LINE

Sets or queries the line number to send to the spectrum display.

Conditions Measurement views: Spectrogram

Group Trace commands

Syntax
```
TRACe:SGRam:SELect:LINE <number>
TRACe:SGRam:SELect:LINE?
```

Related Commands TRACe<x>:SPECtrum

Arguments <number> ::= <NR1> specifies the line number to send to the spectrum display. Range: 0 to the maximum line number of the spectrogram displayed on screen.

Examples `TRACE:SGRAM:SELECT:LINE 75` selects Line #75 in the spectrogram to send to the spectrum display.

TRACe:SPURious:COUNt

Sets or queries how many acquisitions run in the single acquisition mode for multi-trace functions (Max Hold and Average) in the Spurious measurement. This command is effective when TRACe:SPURious:FUNCtion is set to MAXHold or AVERage and INITiate:CONTinuous is set to OFF.

Conditions Measurement views: Spurious

Group Trace commands

Syntax
```
TRACe:SPURious:COUNt <number>
TRACe:SPURious:COUNt?
```

Arguments <number> :: <NR1> specifies the count for multi-trace functions. Range: 1 to 10000.

Examples `TRACE:SPURIOUS:COUNT32` sets the count to 32 for multi-trace functions.

TRACe:SPURious:COUNt:ENABle

Determines whether to enable or disable the count for multi-trace functions (Max Hold and Average) in the Spurious view. This command is effective when TRACe:SPURious:FUNCtion is set to MAXHold or AVERage.

Conditions Measurement views: Spurious

Group Trace commands

Syntax
```
TRACe:SPURious:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe:SPURious:COUNt:ENABle?
```

Arguments `OFF` or `0` disables the count for multi-trace functions.

`ON` or `1` enables the count for multi-trace functions.

Examples `TRACE:SPURIOUS:COUNT:ENABLEON` enables the count for multi-trace functions.

TRACe:SPURious:COUNt:RESet (No Query Form)

Clears the multi-function (Max Hold or Average) data and counter, and restarts the process in the Spurious view. This command is effective when TRACe:SPURious:FUNCtion is set to MAXHold or AVERage.

Conditions Measurement views: Spurious

Group Trace commands

Syntax
```
TRACe:SPURious:COUNt:RESet
```

Arguments None

Examples `TRACE:SPURIOUS:COUNT:RESET` clears the multi-function data and counter, and restarts the process.

TRACe:SPURious:FREeze

Determines whether or not to freeze the display of the trace in the Spurious view.

Conditions Measurement views: Spurious

Group Trace commands

Syntax

```
TRACe:SPURious:FREeze { OFF | ON | 0 | 1 }
TRACe:SPURious:FREeze?
```

Arguments

OFF or 0 updates the display of the trace normally.

ON or 1 stops updating the display of the trace.

Examples

TRACE:SPURIOUS:FREEZEON stops updating the display of the trace.

TRACe:SPURious:FUNCtion

Sets or queries the trace function in the Spurious view.

Conditions

Measurement views: Spurious

Group

Trace commands

Syntax

```
TRACe:SPURious:FUNCtion { NONE | MAXHold | AVERage | AVGLog
}
TRACe:SPURious:FUNCtion?
```

Arguments

NONE selects normal display.

MAXHold selects the Max Hold display that indicates the maximum amplitude drift at each frequency point.

AVERage selects the Average display that indicates the average amplitude drift at each frequency point.

AVGLog selects the Average of logs display that indicates the average amplitude drift of logs at each frequency point.

Examples

TRACE:SPURIOUS:FUNCTIONMAXHold displays the Max Hold trace in the Spurious measurement.

TRACe:WLAN:FLATness:FREeze

Sets or queries whether to freeze the average trace on the WLAN Spectral Flatness display. When the freeze state is turned on, the selected trace stops updating.

NOTE. *WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Trace commands

Syntax
```
TRACe:WLAN:FLATness:FREeze {0|1|OFF|ON}
TRACe:WLAN:FLATness:FREeze?
```

Arguments ON or 1 turns on the freeze state, causing the selected trace to stop updating.

OFF or 0 turns off the freeze state.

Returns Outputs the frozen status of the selected trace as an integral value. 1 indicates a frozen trace and 0 a non-frozen trace.

Examples `TRAC:WLAN:FLAT:FRE 1` freezes the trace, causing it to stop updating.

TRACe:WLAN:FLATness:SHOW

Displays or hides the trace on the WLAN Spectral Flatness display.

NOTE. *WLAN Spectral Flatness is only available for OFDM (non-802.11b) signals.*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Trace commands

Syntax
```
TRACe:WLAN:FLATness:SHOW {0|1|OFF|ON}
TRACe:WLAN:FLATness:SHOW?
```

Arguments ON or 1 displays the trace.

OFF or 0 hides the trace.

Examples `TRACE:WLAN:FLAT:SHOW 1` turns on the display of the trace.

TRACe1:TOVerview

Enables display of or queries the display status of the specified trace. Only Trace1 is valid.

Conditions Measurement views: General Waveform display

Group Trace commands

Syntax
```
TRACe1:TOVerview { OFF | ON | 0 | 1 }
TRACe1:TOVerview?
```

Arguments `OFF` or `0` disables the trace overview.

`ON` or `1` enables the trace overview.

Examples `TRACe1:TOVerview ON` enables display of Trace1.

TRACe1:TOVerview:AVERage:COUNt

Sets or queries the number of traces averaged to generate the specified trace. Only Trace1 is valid.

Conditions Measurement views: Trace Function set to Average

Group Trace commands

Syntax
```
TRACe1:TOVerview:AVERage:COUNt <value>
TRACe1:TOVerview:AVERage:COUNt?
```

Arguments `<value> :: <NR1>` an integer number of traces to average to create the waveform display.

Examples `TRACe1:TOVerview:AVERage:COUNt 200` sets the Average count for Trace 1 to 200.

TRACe1:TOVerview:COUNt

Enables or queries the count set for the specified trace. Only Trace1 is valid.

Conditions Measurement views: Trace Function set to Average

Group Trace commands

Syntax
```
TRACe1:TOVerview:COUNt <value>
TRACe1:TOVerview:COUNt?
```

Arguments `<value> :: <NRf>`

Examples TRACe1:TOVerview:COUNt 200 sets the Trace1 Average count to 200.

TRACe1:TOVerview:COUNt:ENABle

Enables or queries the Average count for the specified trace. Only Trace1 is valid.

Conditions Measurement views: Trace Function set to Average

Group Trace commands

Syntax
```
TRACe1:TOVerview:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe1:TOVerview:COUNt:ENABle?
```

Arguments OFF or 0 disables the trace count.

ON or 1 enables the trace count.

Examples TRACe1:TOVerview:COUNt:ENABle ON enables the Trace1 Average count.

TRACe1:TOVerview:COUNt:RESet (No Query Form)

Resets the waveform count for the specified trace. Only Trace1 is valid..

Conditions Measurement views: Trace Function set to Average

Group Trace commands

Syntax `TRACe1:TOVerview:COUNt:RESet`

Arguments None

Examples `TRACe1:TOVerview:COUNt:RESet` sets the trace count to 1.

TRACe1:TOVerview:DETection

Enables or queries the type of detection for the specified trace. Only Trace1 is valid.

Conditions Measurement views: all

Group Trace commands

Syntax
```
TRACe1:TOVerview:DETection { AVERage | POSitive | NEGative |
POSNegative | SAMPle }
TRACe1:TOVerview:DETection?
```

Arguments `AVERage` displays the average data value for each pixel.

`POSitive` displays the maximum data value for each pixel.

`NEGative` displays the minimum data value for each pixel.

`POSNegative` displays the maximum and minimum data values for each pixel.

`SAMPle` displays the first sample value received for each pixel.

Examples `TRACe1:TOVerview:DETection POSitive` enables positive detection on Trace1.

TRACe1:TOVerview:FREeze

Enables or queries a halt to acquisition updates for the specified trace. Only Trace1 is valid.

Conditions Measurement views: all

Group Trace commands

Syntax

```
TRACe1:TOVerview:FREeze { OFF | ON | 0 | 1 }
TRACe1:TOVerview:FREeze?
```

Arguments

OFF or 0 disables the trace freeze function.

ON or 1 enables the trace freeze function.

Examples

TRACe1:TOVerview:FREeze ON Halts acquisition updates to Trace1.

TRACe1:TOVerview:FUNCTion

Enables or queries the selected Function for the specified trace. Only Trace1 is valid.

Conditions Measurement views: all

Group Trace commands

Syntax

```
TRACe1:TOVerview:FUNCTion { NORMal | AVERage | MAXHold |
MINHold }
TRACe1:TOVerview:FUNCTion?
```

Arguments

NORMal Each new trace is displayed and then replaced by the next trace..

AVERage Multiple traces are averaged together to generate the displayed trace..

MAXHold Displays the maximum value in the trace record for each display point.

MINHold Displays the minimum value in the trace record for each display point.

Examples

TRACe1:TOVerview:FUNCTion AVERage enables the averaging of multiple traces on Trace1.

TRACe<x>:{AM|FM|PM}

Determines whether or not to show the trace in the AM/FM/PM view.

Conditions Measurement views: AM, FM, PM

Group Trace commands

Syntax

```
TRACe<x>:{AM|FM|PM} { OFF | ON | 0 | 1 }
TRACe<x>:{AM|FM|PM}?
```

Arguments

OFF or 0 hides the trace in the AM, FM, or PM view.

ON or 1 shows the trace in the AM, FM, or PM view.

Examples

TRACE:AM ON shows the trace in the AM view.

TRACe<x>:AVTime

Determines whether or not to show the specified trace in the Amplitude versus Time view.

The parameter <x> = 1 to 4; All traces are valid.

Conditions

Measurement views: Amplitude versus Time

Group

Trace commands

Syntax

```
TRACe<x>:AVTime { OFF | ON | 0 | 1 }
TRACe<x>:AVTime?
```

Arguments

OFF or 0 hides the specified trace in the Amplitude versus Time view.

ON or 1 shows the specified trace in the Amplitude versus Time view.

Examples

TRACE1:AVTIME ON shows Trace 1 in the Amplitude versus Time view.

TRACe<x>:AVTime:AVERage:COUNt

Sets or queries the number of traces to combine. This command is effective when you select AVERage with the TRACe<x>:AVTime:FUNCtion command.

The parameter <x> = 1 to 3; Trace 4 (math trace) is invalid.

Conditions

Measurement views: Amplitude versus Time

Group

Trace commands

Syntax

```
TRACe<x>:AVTime:AVERage:COUNt <number>
TRACe<x>:AVTime:AVERage:COUNt?
```

Arguments

<number> :: <NR1> specifies the number of traces to combine for averaging. Range: 1 to 10000.

Examples

TRACE1:AVTIME:AVERAGE:COUNT 64 sets the average count to 64 for Trace 1.

TRACe<x>:AVTime:AVERage:RESet (No Query Form)

Restarts acquisition and display of waveforms for the specified trace. For an Average, Max Hold, or Min Hold trace, it restarts the sequence, discarding accumulated data and resetting the counter.

The parameter <x> = 1 to 3; Trace 4 (math trace) is invalid.

Conditions

Measurement views: Amplitude versus Time

Group

Trace commands

Syntax

```
TRACe<x>:AVTime:AVERage:RESet
```

Related Commands

TRACe<x>:AVTime:FUNCtion

Arguments

None

Examples

TRACE1:AVTIME:AVERAGE:RESET restarts acquisition and display of waveforms for Trace 1.

TRACe<x>:AVTime:COUNt

Sets or queries how many acquisitions run in the single acquisition mode for the Max or Min Hold trace in the Amplitude versus Time measurement. This command is effective when TRACe<x>:AVTime:FUNCtion is set to MAXHold or MINHold and INITiate:CONTinuous is set to OFF.

The parameter <x> = 1 to 3; Trace 4 (math trace) is invalid.

Conditions

Measurement views: Amplitude versus Time

Group Trace commands

Syntax
```
TRACe<x>:AVTime:COUNt <number>
TRACe<x>:AVTime:COUNt?
```

Arguments <number> :: <NR1> specifies the count for Max/Min Hold. Range: 1 to 10000.

Examples TRACE1:AVTIME:COUNT 32 sets the count to 32 for Trace 1.

TRACe<x>:AVTime:COUNt:ENABle

Determines whether to enable or disable the count for the Max or Min Hold trace in the Amplitude versus Time view. This command is effective when TRACe<x>:AVTime:FUNCtion is set to MAXHold or MINHold.

The parameter <x> = 1 to 3; Trace 4 (math trace) is invalid.

Conditions Measurement views: Amplitude versus Time

Group Trace commands

Syntax
```
TRACe<x>:AVTime:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe<x>:AVTime:COUNt:ENABle?
```

Arguments OFF or 0 disables the count for the Max/Min Hold trace.

ON or 1 enables the count for the Max/Min Hold trace.

Examples TRACE1:AVTIME:COUNT:ENABLE ON enables the Max/Min Hold count for Trace 1.

TRACe<x>:AVTime:COUNt:RESet (No Query Form)

Clears the Max or Min Hold data and counter, and restarts the process for the specified trace in the Amplitude versus Time view. This command is effective when TRACe<x>:AVTime:FUNCtion is set to MAXHold or MINHold.

The parameter <x> = 1 to 3; Trace 4 (math trace) is invalid.

Conditions Measurement views: Amplitude versus Time

Group Trace commands

Syntax `TRACe<x>:AVTime:COUNt:RESet`

Arguments None

Examples `TRACE1:AVTIME:COUNT:RESET` clears the Max/Min Hold data and counter, and restarts the process for Trace 1.

TRACe<x>:AVTime:DETection

Sets or queries the display detector (method to be used for decimating traces to fit the available horizontal space on screen). The number of horizontal pixels on screen is generally smaller than that of waveform data points. When actually displayed, the waveform data is therefore thinned out, according to the number of pixels, for being compressed.

Conditions Measurement views: Amplitude versus Time

Group Trace commands

Syntax
```
TRACe<x>:AVTime:DETection { AVERage | POSitive | NEGative |
POSNegative | SAMPle }
TRACe<x>:AVTime:DETection?
```

Arguments `AVERage` displays the average data value for each pixel.

`POSitive` displays the maximum data value for each pixel.

`NEGative` displays the minimum data value for each pixel.

`POSNegative` displays the maximum and minimum data values for each pixel.

`SAMPle` displays the first sample value received for each pixel.

Examples `TRACE:AVTIME:DETECTION AVERage` specifies that the trace displays the average data value for each pixel.

TRACe<x>:AVTime:FREeze

Determines whether or not to freeze the display of the specified trace in the Amplitude versus Time view.

The parameter <x> = 1 to 4; All traces are valid.

Conditions Measurement views: Amplitude versus Time

Group Trace commands

Syntax

```
TRACe<x>:AVTime:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:AVTime:FREeze?
```

Arguments `OFF` or `0` updates the display of the specified trace normally.

`ON` or `1` stops updating the display of the specified trace.

Examples `TRACE1:AVTIME:FREEZE ON` freezes the display for Trace 1.

TRACe<x>:AVTime:FUNCtion

Sets or queries the function for the specified trace in the Amplitude versus Time view.

The parameter <x> = 1 to 3; Trace 4 (math trace) is invalid.

Conditions Measurement views: Amplitude versus Time

Group Trace commands

Syntax

```
TRACe<x>:AVTime:FUNCtion { NORMal | AVERage | MAXHold |
MINHold }
TRACe<x>:AVTime:FUNCtion?
```

Arguments `NORMal` selects the normal display.

`AVERage` selects the Average display that indicates the average amplitude at each time point.

`MAXHold` selects the Max Hold display that indicates the maximum amplitude at each time point.

MINHold selects the Min Hold display that indicates the minimum amplitude at each time point.

Examples TRACE1:AVTIME:FUNCTION MAXHold selects Max Hold for Trace 1 in the Amplitude versus Time view.

TRACe<x>:AVTime:LEFToperand

Sets or queries the left operand for the math trace (Trace 4) in the Amplitude versus Time view.

The parameter <x> = 4; Only Trace 4 (math trace) is valid.

Conditions Measurement views: Amplitude versus Time

Group Trace commands

Syntax
```
TRACe<x>:AVTime:LEFToperand { TRACE1 | TRACE2 | TRACE3 }
TRACe<x>:AVTime:LEFToperand?
```

Related Commands TRACe<x>:AVTime:RIGHtoperand

Arguments TRACE1 selects Trace 1 as the left operand for the math trace.

TRACE2 selects Trace 2 as the left operand for the math trace.

TRACE3 selects Trace 3 as the left operand for the math trace.

Examples TRACE4:AVTIME:LEFTOPERAND TRACE2 selects Trace 2 as the left operand for the math trace.

TRACe<x>:AVTime:RIGHtoperand

Sets or queries the right operand for the math trace (Trace 4) in the Amplitude versus Time view.

The parameter <x> = 4; Only Trace 4 (math trace) is valid.

Conditions Measurement views: Amplitude versus Time

Group Trace commands

Syntax

```
TRACe<x>:AVTime:RIGHtoperand { TRACE1 | TRACE2 | TRACE3 }
TRACe<x>:AVTime:RIGHtoperand?
```

Related Commands TRACe<x>:AVTime:LEFToperand

Arguments

`TRACE1` selects Trace 1 as the right operand for the math trace.

`TRACE2` selects Trace 2 as the right operand for the math trace.

`TRACE3` selects Trace 3 as the right operand for the math trace.

Examples `TRACE4:AVTIME:RIGHTOPERAND TRACE1` selects Trace 1 as the right operand for the math trace.

TRACe<x>:AVTime:SELect

Sets or queries the trace whose measurement results are being displayed in the readout on the top and bottom of the view.

The parameter <x> = 1 to 4; All traces are valid.

Conditions Measurement views: Amplitude versus Time

Group Trace commands

Syntax

```
TRACe<x>:AVTime:SELect
TRACe<x>:AVTime:SELect?
```

Arguments None

Returns `{ 0 | 1 }`

`0` indicates that the results are not being displayed in the readout for the trace.

`1` indicates that the results are being displayed in the readout for the trace.

Examples `TRACE1:AVTIME:SELECT` selects Trace 1 to display the measurement results in the readout.

`TRACE1:AVTIME:SELECT?` might return `1`, indicating that the results are being displayed in the readout for Trace 1.

TRACe<x>:BLUEtooth:CONSte

Sets or queries the specified trace to display or hide in the Bluetooth Constellation display.

The parameter <x> is trace 1 or 2.

Conditions Measurement views: Bluetooth Constellation

Group Trace commands

Syntax

```
TRACe<x>:BLUEtooth:CONSte { OFF | ON | 0 | 1 }
TRACe<x>:BLUEtooth:CONSte?
```

Arguments OFF or 0 hides the selected trace.
ON or 1 displays the selected trace.

Returns 0 indicates the specified trace is not shown in the Bluetooth Constellation display.

1 indicates the specified trace is shown in the Bluetooth Constellation display.

Examples TRACe1:BLUEtooth:CONSte ON specifies that trace 1 appears in the Bluetooth Constellation display.

TRACe<x>:BLUEtooth:CONSte:FREeze

Sets or queries the specified trace to freeze or not to freeze in the Bluetooth Constellation display.

The parameter <x> is trace 1 or 2.

Conditions Measurement views: Bluetooth Constellation

Group Trace commands

Syntax

```
TRACe<x>:BLUEtooth:CONSte:FREeze {OFF | ON | 0 | 1 }
TRACe<x>:BLUEtooth:CONSte:FREeze?
```

Arguments OFF or 0 updates the display of the selected trace normally.
ON or 1 stops updating the display of the selected trace.

Returns 0 indicates the specified trace is being updated in the Bluetooth Constellation display.
1 indicates the specified trace is not being updated in the Bluetooth Constellation display.

Examples TRACe1:BLUEtooth:CONSte:FREeze ON freezes the display for trace 1 in the Bluetooth Constellation display.

TRACe<x>:BLUEtooth:CONSte:MODE

Sets or queries the Bluetooth Constellation trace display mode.

The parameter <x> is trace 1 or 2.

Conditions Measurement views: Bluetooth Constellation

Group Trace commands

Syntax
```
TRACe<x>:BLUEtooth:CONSte:MODE { VECT | SYMB| LIN }
TRACe<x>:BLUEtooth:CONSte:MODE?
```

Arguments VECT connects adjacent symbol pints with the signal locus.
SYMB displays individual symbol points.
LIN displays linear symbol points.

Examples TRACe1:BLUEtooth:XONSte:MODE VECT shows the Bluetooth Constellation display is connecting adjacent symbol points with the signal locus.

TRACe<x>:BLUEtooth:CONSte:SELect

Sets or queries the active trace in the Bluetooth Constellation diagram.

The parameter <x> is trace 1 or 2.

Conditions Measurement views: Bluetooth Constellation

Group Trace commands

Syntax
```
TRACe<x>:BLUEtooth:CONSte:SELect <active>
TRACe<x>:BLUEtooth:CONSte:SELect?
```

Arguments <active> ::= {0 | 1}

Where:
0, the specified trace is not the selected trace.
1, the specified trace is the selected trace.

Examples TRACe1:BLUEtooth:CONSte:SELect specifies trace 1 as the selected trace.

TRACe<x>:BLUEtooth:EDIagram:ENABle:I

Determines whether to show or hide the I trace in the Bluetooth eye diagram.

Conditions Measurement views: Bluetooth Eye diagram

Group Trace commands

Syntax
```
TRACe<x>:BLUEtooth:EDIagram:ENABle:I { OFF | ON | 0 | 1 }
TRACe<x>:BLUEtooth:EDIagram:ENABle:I?
```

Arguments OFF or 0 hides the I trace.
ON or 1 shows the I trace.

Examples `TRACEX:BLUETOOTH:EDIAGRAM:ENABLE:I` ON shows the I trace in the eye diagram.

TRACe<x>:BLUEtooth:EDIagram:ENABle:Q

Determines whether to show or hide the Q trace in the Bluetooth Eye diagram.

Conditions Measurement views: Bluetooth Eye diagram

Group Trace commands

Syntax
```
TRACe<x>:BLUEtooth:EDIagram:ENABle:Q { OFF | ON | 0 | 1 }
TRACe<x>:BLUEtooth:EDIagram:ENABle:Q?
```

Arguments OFF or 0 hides the Q trace.
ON or 1 shows the Q trace.

Examples `TRACEX:BLUETOOTH:EDIAGRAM:ENABLE:Q` ON shows the Q trace in the eye diagram.

TRACe<x>:BLUEtooth:EDIagram:SELect:I

Selects the I trace in the Bluetooth eye diagram. The query version of this command returns whether the I trace is selected or not.

Conditions Measurement views: Bluetooth Eye diagram

Group Trace commands

Syntax
```
TRACe<x>:BLUEtooth:EDIagram:SELect:I { 0 | 1 }
TRACe<x>:BLUEtooth:EDIagram:SELect:I?
```

Arguments None

Returns 0 indicates that the I trace is deselected.
1 indicates that the I trace is selected.

Examples `TRACEX:BLUETOOTH:EDIAGRAM:SELECT:I` selects the I trace in the eye diagram.

TRACe<x>:BLUEtooth:EDIagram:SELect:Q

Selects the Q trace in the Bluetooth eye diagram. The query version of this command returns whether the I trace is selected or not.

Conditions Measurement views: Bluetooth Eye diagram

Group Trace commands

Syntax
```
TRACe<x>:BLUEtooth:EDIagram:SELect:Q { 0 | 1 }
TRACe<x>:BLUEtooth:EDIagram:SELect:Q?
```

Arguments None

Returns 0 indicates that the Q trace is deselected.
1 indicates that the Q trace is selected.

Examples `TRACEX:BLUETOOTH:EDIAGRAM:SELECT:Q` selects the Q trace in the eye diagram.

TRACe<x>:BLUEtooth:FDVTime:MODE

Sets or queries whether to display the Frequency Deviation vs. Time trace as vectors or symbols (points).

Conditions Measurement views: Frequency Deviation vs. Time

Group Trace commands

Syntax
```
TRACe<x>:BLUEtooth:FDVTime:MODE { VECTors | SYMBols }
TRACe<x>:BLUEtooth:FDVTime:MODE?
```

Arguments VECTors connects adjacent symbol points with lines.
SYMBols displays individual symbol points.

Examples `TRACe:BLUEtooth:FDVTime:MODE VECT` sets the display to connect points in the display with lines.

TRACe<x>:CCDF:FREeze

Determines whether or not to freeze the display of the specified trace (Trace 1 or 2) in the CCDF view.

The parameter <x> = 1 or 2; Trace 3 (Gaussian curve) is invalid.

Conditions Measurement views: CCDF

Group Trace commands

Syntax
```
TRACe<x>:CCDF:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:CCDF:FREeze?
```

Arguments OFF or 0 updates the display of the specified trace normally.

ON or 1 stops updating the display of the specified trace.

Examples TRACE1:CCDF:FREEZE ON freezes the display for Trace 1.

TRACe<x>:CCDF:SELect

Sets or queries the trace whose measurement results are being displayed in the readout on the top of the view. The selected trace is indicated by the measurement pointer (pink triangle) on the waveform.

The parameter <x> = 1 to 3; All traces are valid.

Conditions Measurement views: CCDF

Group Trace commands

Syntax
```
TRACe<x>:CCDF:SELect
TRACe<x>:CCDF:SELect?
```

Arguments None

Returns { 0 | 1 }

0 indicates that the results are not being displayed in the readout for the trace.

1 indicates that the results are being displayed in the readout for the trace.

Examples TRACE1:CCDF:SELECT selects Trace 1 to display the measurement results in the readout.

TRACE1:CCDF:SELECT? might return 1, indicating that the results are being displayed in the readout for Trace 1.

TRACe<x>:CCDF:SHOW

Determines whether to show or hide the specified trace in the CCDF view.

The parameter <x> = 1 to 3; All traces are valid.

Conditions Measurement views: CCDF

Group Trace commands

Syntax
```
TRACe<x>:CCDF:SHOW { OFF | ON | 0 | 1 }
TRACe<x>:CCDF:SHOW?
```

Arguments `OFF` or `0` hides the specified trace.

`ON` or `1` shows the specified trace.

Examples `TRACE1:CCDF:SHOW ON` shows Trace 1 in the CCDF view.

TRACe<x>:CCDF:X

Sets or queries the horizontal position of the measurement pointer (pink triangle) to measure the CCDF. Use the TRACe<x>:CCDF:Y? query to read the value.

The parameter <x> = 1 to 3.

NOTE. *Use the TRACe<x>:CCDF:SHOW command to show the specified trace and the TRACe<x>:CCDF:SELect command to select the trace before running the TRACe<x>:CCDF:X command.*

Conditions Measurement views: CCDF

Group Trace commands

Syntax
```
TRACe<x>:CCDF:X <value>
TRACe<x>:CCDF:X?
```

Arguments `<value> ::= <NRf>` specifies the horizontal position of the measurement pointer.
Range: 0 to 20 dB.

Examples `TRACE1:CCDF:X 5` puts the measurement pointer at 5 dB on Trace 1.

TRACe<x>:CCDF:Y? (Query Only)

Queries the vertical position (CCDF value) of the measurement pointer (displayed as a pink triangle). Use the TRACe<x>:CCDF:X command to set the horizontal position of the pointer.

The parameter <x> = 1 to 3.

NOTE. *Use the TRACe<x>:CCDF:SHOW command to show the specified trace and the TRACe<x>:CCDF:SELect command to select the trace before running the TRACe<x>:CCDF:Y? query.*

Conditions Measurement views: CCDF

Group Trace commands

Syntax `TRACe<x>:CCDF:Y?`

Arguments None

Returns `<value> ::  <NRf>` is the vertical position (CCDF) of the measurement pointer.
Range: 0 to 100%.

The value of 99.0999953003E+36 is returned if the trace is not available.

Examples `TRACE1:CCDF:Y?` might return `14.72`, indicating the CCDF is 14.72% at the measurement pointer on Trace 1.

TRACe<x>:CONSte

Specifies whether or not to show the specified trace (Trace 1 or 2) in the Constellation display.

The parameter <x> = 1 or 2.

Conditions Measurement views: Constellation

Group Trace commands

Syntax

```
TRACe<x>:CONSte {OFF | ON | 0 | 1 }
TRACe<x>:CONSte?
```

Arguments

OFF or 0 hides the selected trace.

ON or 1 displays the selected trace.

Returns

{ 0 } Indicates the specified trace is not shown in the Constellation display.

{ 1 } Indicates the specified trace is shown in the Constellation display.

Examples

TRACE1:CONSTE 2 specifies that Trace 2 be shown in the Constellation display.

TRACe<x>:CONSte:FREeze

Specifies whether or not to freeze (halt updates) to the display of the specified trace (Trace 1 or 2) in the Constellation display.

The parameter <x> = 1 or 2.

Conditions

Measurement views: Constellation

Group

Trace commands

Syntax

```
TRACe<x>:CONSte:FREeze {OFF | ON | 0 | 1 }
TRACe<x>:CONSte:FREeze?
```

Arguments

OFF or 0 updates the display of the specified trace normally.

ON or 1 stops updating the display of the specified trace.

Returns

{ 0 } Indicates the specified trace is being updated.

{ 1 } Indicates the specified trace is not being updated.

Examples

TRACE1:CONSTE:FREEZE ON freezes the display for Trace 1.

TRACe<x>:CONSte:MODE

Sets or queries how to display the constellation trace.

Conditions Measurement views: Constellation

Group Trace commands

Syntax
```
TRACe<x>:CONSte:MODE { VECTors | SYMBols }
TRACe<x>:CONSte:MODE?
```

Arguments `VECTors` connects adjacent symbol points with the signal locus.

`SYMBols` displays individual symbol points.

Examples `TRACE<X>:CONSTE:MODE VECTors` shows the constellation connecting adjacent symbol points with the signal locus.

TRACe<x>:CONSte:Q:OFFSet

Sets the Q offset in the Constellation display to Use shared Pref, Remove Q offset, or Include Q offset.

The parameter <x> = 1 or 2.

Conditions Measurement views: Constellation

Group Trace commands

Syntax
```
TRACe<x>:CONSte:Q:OFFSet { SHARed | RQOFset | IQOFset }
TRACe<x>:CONSte:Q:OFFSet?
```

Arguments `SHARed` means that the Q offset applied to the selected trace will be the same as the Remove Q offset setting located on the Prefs tab (which applies to all GP Digital Modulation displays).

`Remove Q offset` removes the Q offset, possibly overriding the Remove Q offset setting on the Prefs tab. Changing the Constellation display Q Offset value with this command does not affect the Q offset setting on other displays.

`Include Q offset` includes the Q offset, possibly overriding the Remove Q offset setting on the Prefs tab. Changing the Constellation display Q Offset value with this command does not affect the Q offset setting on other displays.

Returns `{ SHAR }` indicates that Q Offset is set to Use Shared Pref.

`{ RQOF }` indicates that Q Offset is set to Remove Q Offset.

`{ IQOF }` indicates that Q Offset is set to Include Q Offset.

Examples `TRACE1:CONSTE:OFFSET SHAR` sets the Constellation display Q Offset to Use Shared Pref.

TRACe<x>:CONSte:SELect (No Query Form)

Sets or queries the trace to which the Show, Freeze, Content, and Q Offset settings apply.

The parameter <x> = 1 or 2.

Conditions Measurement views: Constellation

Group Trace commands

Syntax
```
TRACe<x>:CONSte:SELect
TRACe<x>:CONSte:SELect?
```

Arguments None

Returns `{ 0 }` indicates the specified trace is not the selected trace.

`{ 1 }` indicates the specified trace is the selected trace.

Examples `TRACE1:CONSTE:SELECT` specifies Trace 1 as the selected trace.

TRACe<x>:DPX

Determines whether or not to show the specified trace in the DPX, Spectrum view.

The parameter <x> = 1 to 7; All traces are valid. Trace 6 is for the DPXogram trace and Trace 7 is for the Ogram line.

Conditions Measurement views: DPX, Spectrum

Group Trace commands

Syntax

```
TRACe<x>:DPX { OFF | ON | 0 | 1 }
TRACe<x>:DPX?
```

Arguments

OFF or 0 hides the specified trace in the DPX, Spectrum view.

ON or 1 shows the specified trace in the DPX, Spectrum view.

Examples

TRACE1:DPX ON shows Trace 1 (the maximum trace) in the DPX, Spectrum view.

TRACe<x>:DPX:AVERage:COUNt

Sets or queries the number of traces to combine for averaging in the DPX, Spectrum view.

The parameter <x> = 3; Only Trace 3 (average trace) is valid.

Conditions

Measurement views: DPX, Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:DPX:AVERage:COUNt <number>
TRACe<x>:DPX:AVERage:COUNt?
```

Arguments

<number> :: <NR1> specifies the number of traces to combine for averaging. Range: 1 to 10000.

Examples

TRACE3:DPX:AVERAGE:COUNT 32 sets the average count to 32.

TRACe<x>:DPX:COLor:CURVe

Sets or queries how colors are mapped to the signal density in the DPX, Spectrum bitmap display. The mapping can be linear (Curve = 1), or it can be set to concentrate the resolution on the lower level of the range (Curve > 1) or the mapping can be set to show the best resolution on the upper range of density or hit count (Curve = 0.1 – 0.99). This command is only valid for the Bitmap display.

The parameter <x> = 5; only Trace5 is valid.

Conditions

Measurement views: DPX, Spectrum

Group Trace commands

Syntax
```
TRACe<x>:DPX:COLor:CURVe <value>
TRACe<x>:DPX:COLor:CURVe?
```

Arguments <value> :: <NRf> specifies how colors are mapped to the signal density. Range:

Examples TRACe<x>:DPX:COLor:CURVe 1.5 concentrates the resolution on the lower level of the range on the Bitmap display.

TRACe<x>:DPX:COLor:INTensity

Sets or queries the color intensity in the DPX, Spectrum view. The value is common to all traces.

The parameter <x> = 1 to 5; All traces are valid.

Conditions Measurement views: DPX, Spectrum

Group Trace commands

Syntax
```
TRACe<x>:DPX:COLor:INTensity <value>
TRACe<x>:DPX:COLor:INTensity?
```

Arguments <value> :: <NRf> specifies color intensity. Range: 1 to 100%.

Examples TRACE1:DPX:COLOR:INTENSITY 30 sets the color intensity to 30%.

TRACe<x>:DPX:COLor:SCALe:AUTO (No Query Form)

Automatically adjusts the Max and Min color settings to display the broadest range of colors in the DPX, Spectrum bitmap display. This command is only valid for the Bitmap display.

The parameter <x> = 5; only Trace5 is valid.

Conditions Measurement views: DPX, Spectrum

Group Trace commands

Syntax `TRACe<x>:DPX:COLor:SCALe:AUTO`

Arguments `<x>::=5` specifies the auto-color setting for the Bitmap trace.

Examples `TRACe5:DPX:COLor:SCALe:AUTO` autosets the Bitmap display color scheme.

TRACe<x>:DPX:DETection

Sets or queries the detection type (POSitive, NEGative, AVERage) for traces 1, 2 and 3 in the DPX, Spectrum view.

Trace 4 (math trace) and Trace 5 (bitmap trace) are invalid choices. Trace 6 is only valid for DPXogram.

Conditions Measurement views: DPX, Spectrum

Group Trace commands

Syntax
```
TRACe<x>:DPX:DETection { AVERage | NEGative | POSitive }
TRACe<x>:DPX:DETection?
```

Arguments The following table shows the trace function and display. For the average trace, use the TRACe<x>:DPX:AVERage:COUNt command to set the average count.

Examples `TRACE1:DPX:DETection AVERage` enables analysis to look for AVERage on Trace 1.

TRACe<x>:DPX:DGRAM:SELect:LINE

Sets or queries a line from the DPXogram display to send to the DPX, Spectrum display.

Trace 6 is the DPXogram trace. No other traces are allowed.

Conditions Measurement views: DPX, Spectrum

Group Trace commands

Syntax

```
TRACe<x>:DPX:DGRAM:SELect:LINE <number>
TRACe<x>:DPX:DGRAM:SELect:LINE?
```

Arguments

<number> ::= <NR1> specifies the line number in the DPXogram display.

Examples

TRACE6:DPX:DGRAM:SELECT:LINE 75 selects line #75 in the DPXogram display to send to the DPX, Spectrum display.

TRACe<x>:DPX:DOT:PERSistent

Determines whether to enable or disable the dot persistence for the bitmap trace (Trace 5) in the DPX, Spectrum view.

The parameter <x> = 5; Only Trace 5 (bitmap trace) is valid.

Conditions

Measurement views: DPX, Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:DPX:DOT:PERSistent { OFF | ON | 0 | 1 }
TRACe<x>:DPX:DOT:PERSistent?
```

Arguments

`OFF` or `0` disables the dot persistence.

`ON` or `1` enables the dot persistence.

Examples

`TRACE5:DPX:DOT:PERSISTENT ON` enables the dot persistence in the DPX, Spectrum view.

TRACe<x>:DPX:DOT:PERSistent:TYPE

Sets or queries the persistence type for the bitmap trace (Trace 5) in the DPX, Spectrum view.

The parameter <x> = 5; Only Trace 5 (bitmap trace) is valid.

Conditions

Measurement views: DPX, Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:DPX:DOT:PERSistent:TYPE { VARiable | INFinite }
TRACe<x>:DPX:DOT:PERSistent:TYPE?
```

Arguments

`VARiable` selects the variable persistence display which leaves acquired data points on the display for a period of time specified by the TRACe<x>:DPX:DOT:PERSistent:VARiable command.

`INFinite` selects the infinite persistence display which accumulates data points on the display indefinitely.

Examples

`TRACE5:DPX:DOT:PERSISTENT:TYPE VARiable` selects the variable persistence display.

TRACe<x>:DPX:DOT:PERSistent:VARiable

Sets or queries how long data points are displayed. This command is effective when TRACe<x>:DPX:DOT:PERSistent:TYPE is set to VARiable. This affects the display only.

The parameter <x> = 5; Only Trace 5 (bitmap trace) is valid.

Conditions

Measurement views: DPX, Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:DPX:DOT:PERSistent:VARiable <number>
TRACe<x>:DPX:DOT:PERSistent:VARiable?
```

Arguments

`<number> ::= <NR1>` specifies the how long waveform points are displayed on the screen. Range: 100 ms to 60 s (the default value is 1.0 s).

Examples

`TRACE5:DPX:DOT:PERSISTENT:VARIABLE 20` specifies that the waveform points are displayed on the screen for a period of 20 before they disappear.

TRACe<x>:DPX:FREeze

Determines whether or not to freeze the display of the specified trace in the DPX, Spectrum view.

The parameter <x> = 1 to 6; All traces are valid. Trace 6 is for the DPXogram trace and Trace 7 is for the Ogram line.

Conditions Measurement views: DPX, Spectrum

Group Trace commands

Syntax
```
TRACe<x>:DPX:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:DPX:FREeze?
```

Arguments `OFF` or `0` updates the display of the specified trace normally.

`ON` or `1` stops updating the display of the specified trace.

Examples `TRACE1:DPX:FREEZE ON` freezes the display for the +peak trace.

TRACe<x>:DPX:FUNCtion

Sets or queries the trace function for the +Peak, -Peak, or Average trace (Trace 1, 2, or 3, respectively) in the DPX, Spectrum view.

The parameter <x> = 1 to 3; Trace 4 (math trace) and Trace 5 (bitmap trace) are invalid.

Conditions Measurement views: DPX, Spectrum

Group Trace commands

Syntax
```
TRACe<x>:DPX:FUNCtion { NORMal | HOLD | AVERage }
TRACe<x>:DPX:FUNCtion?
```

Arguments The following table shows the trace function and display. For the average trace, use the TRACe<x>:DPX:AVERage:COUNt command to set the average count.

Function	Trace1 (+Peak trace)	Trace2 (-Peak trace)	Trace3 (Average trace)
NORMal	Normal spectrum (Detection: +Peak)	Normal spectrum (Detection: -Peak)	Normal spectrum (Detection: Average)
HOLD	Max-hold spectrum	Min-hold spectrum	NA
AVERage	NA	NA	Average spectrum

Examples `TRACE1:DPX:FUNCTION HOLD` selects the max hold waveform for Trace 1.

TRACe<x>:DPX:LEFToperand

Sets or queries the left operand for the math trace (Trace 4) in the DPX, Spectrum view.

The parameter <x> = 4; Only Trace 4 (math trace) is valid.

Conditions Measurement views: DPX, Spectrum

Group Trace commands

Syntax

```
TRACe<x>:DPX:LEFToperand { TRACe1 | TRACe2 | TRACe3 }
TRACe<x>:DPX:LEFToperand?
```

Related Commands TRACe<x>:DPX:RIGHtoperand

Arguments `TRACE1` selects Trace 1 as the left operand for the math trace.

`TRACE2` selects Trace 2 as the left operand for the math trace.

`TRACE3` selects Trace 3 as the left operand for the math trace.

Examples `TRACE4:DPX:LEFTOPERAND TRACE2` selects Trace 2 as the left operand for the math trace.

TRACe<x>:DPX:RIGHtoperand

Sets or queries the right operand for the math trace (Trace 4) in the DPX, Spectrum view.

The parameter <x> = 4; Only Trace 4 (math trace) is valid.

Conditions Measurement views: DPX, Spectrum

Group Trace commands

Syntax

```
TRACe<x>:DPX:RIGHtoperand { TRACE1 | TRACE2 | TRACE3 }
TRACe<x>:DPX:RIGHtoperand?
```

Related Commands TRACe<x>:DPX:LEFToperand

Arguments

`TRACE1` selects Trace 1 as the right operand for the math trace.

`TRACE2` selects Trace 2 as the right operand for the math trace.

`TRACE3` selects Trace 3 as the right operand for the math trace.

Examples

`TRACE4:DPX:RIGHTOPERAND TRACE1` selects Trace 1 as the right operand for the math trace.

TRACe<x>:DPX:SELect

Sets or queries the trace to display the readout at the upper left of the DPX, Spectrum view.

The parameter <x> = 1 to 7; All traces are valid. Trace 6 is for the DPXogram trace and Trace 7 is for the Ogram line.

Conditions

Measurement views: DPX, Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:DPX:SELect
TRACe<x>:DPX:SELect?
```

Arguments

None

Returns

`{ 0 | 1 }`

`0` indicates that the readout is not being displayed for the specified trace.

`1` indicates that the readout is being displayed for the specified trace.

Examples

`TRACE1:DPX:SELECT` selects Trace 1 (+Peak trace) to display the readout.

`TRACE1:DPX:SELECT?` might return 1, indicating that the readout is being displayed for Trace 1.

TRACe<x>:{FSETtling|PSETtling}:FREeze

Determines whether to freeze the specified trace in the Settling Time measurement display.

The parameter <x> = 1 or 2.

Conditions Measurement views: Frequency and Phase Settling Time

Group Trace commands

Syntax
```
TRACe<x>:{FSETtling|PSETtling}:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:{FSETtling|PSETtling}:FREeze?
```

Arguments `OFF` or `0` updates the trace display normally.

`ON` or `1` stops updating trace display.

Examples `TRACE1:FSETTLING:FREEZE ON` stops updating the Trace 1 display.

TRACe<x>:{FSETtling|PSETtling}:SELect

Selects the trace in the settling time measurement. The query returns the currently selected trace.

The parameter <x> = 1 or 2.

Conditions Measurement views: Frequency and Phase Settling Time

Group Trace commands

Syntax
```
TRACe<x>:{FSETtling|PSETtling}:SELect
TRACe<x>:{FSETtling|PSETtling}:SELect?
```

Arguments None

Returns `0` (not selected) or `1` (selected).

Examples `TRACE2:FSETTLING:SELECT` selects Trace 2.

TRACe<x>:{FSETtling|PSETtling}:SHOW

Determines whether to show or hide the specified trace in the settling time view.

The parameter <x> = 1 or 2.

Conditions Measurement views: Frequency and Phase Settling Time

Group Trace commands

Syntax
```
TRACe<x>:{FSETtling|PSETtling}:SHOW { OFF | ON | 0 | 1 }
TRACe<x>:{FSETtling|PSETtling}:SHOW?
```

Arguments `OFF` or `0` hides the specified trace.

`ON` or `1` shows the specified trace.

Examples `TRACE1:PSETTLING:SHOW ON` shows Trace 1 in the Phase Settling Time view.

TRACe<x>:FVTime:DETection

Enables or queries the type of detection for the specified trace. The parameter <x> represents traces 1 to 4.

Conditions Measurement views: Frequency versus Time

Group Trace commands

Syntax
```
TRACe<x>:FVTime:DETection { AVERage | POSitive | NEGative |
POSNegative | SAMPle }
TRACe<x>:FVTime:DETection?
```

Arguments `AVERage` displays the average data value for each pixel.

`POSitive` displays the maximum data value for each pixel.

`NEGative` displays the minimum data value for each pixel.

`POSNegative` displays the maximum and minimum data values for each pixel.

`SAMPle` displays the first sample value received for each pixel.

Examples `TRACe2:FVTime:DETection POSitive` enables positive detection on Trace2.

TRACe<x>:LTE:ACLR:COUNt

Sets or queries how many acquisitions run in the single acquisition mode for multitrace functions (Max Hold and Average) in the LTE ACLR measurement. This command is effective when `TRACe<x>: LTE:ACLR:FUNCtion` is set to `MAXHold` or `AVERage` and `INITiate:CONTinuous` is set to `OFF`.

Conditions Measurement view: LTE ACLR

Group Trace commands

Syntax `TRACe<x>:LTE:ACLR:COUNt <number>`

Related Commands TRACe<x>:LTE:ACLR:FUNCtion

Arguments <number>::<NR1> specifies the count for multi-trace functions. Range: 1 to 10000.

Returns <NR1>

Examples `TRACE1:LTE:ACLR:COUNT 32` sets the count to 32 for multitrace functions.

TRACe<x>:LTE:ACLR:COUNt:RESet (No Query Form)

Clears the multifunction (Max Hold or Average) data and counter and restarts the process in the LTE ACLR display. This command is effective when `TRACe<x>:LTE:ACLR:FUNCtion` is set to `MAXHold` or `AVERage`.

Conditions Measurement view: LTE ACLR

Group Trace commands

Syntax `TRACe<x>:LTE:ACLR:COUNt:RESet`

Related Commands TRACe<x>:LTE:ACLR:FUNCtion

Arguments None

Examples `TRACE1:LTE:ACLR:COUNT:RESET` clears the multifunction data and counter, and restarts the process.

TRACe<x>:LTE:ACLR:FUNCtion

Sets or queries the trace function in the LTE ACLR display.

Conditions Measurement view: LTE ACLR

Group Trace commands

Syntax
```
TRACe<x>:LTE:ACLR:FUNCtion { NONE | MAXH | AVER }
TRACe<x>:LTE:ACLR:FUNCtion?
```

Arguments NONE: sets Normal display of Trace 1.

MAXHold: sets Max Hold display of Trace 1, which indicates the maximum amplitude drift at each frequency point.

AVERage: sets Average display of Trace 1, which indicates the average amplitude drift at each frequency point.

Returns NONE means that Trace 1 is set to Normal display.

MAXHold means Trace 1 is set to Max Hold display.

AVERage means Trace 1 is set to Average display.

Examples `TRACE1:LTE:ACLR:FUNCTION MAXH` displays the Max Hold trace in the LTE ACLR display.

TRACe<x>:LTE:CONSte:SYNChronization:SEQUence:STATe? (Query Only)

Returns whether or not a measurement signal (Primary Synchronization Signal/Secondary Synchronization Signal) is selected for the specified trace in the Trace tab of the settings control panel of the LTE Constellation display.

Conditions Measurement view: LTE Constellation

Group Trace commands

Syntax `TRACe<x>:LTE:CONSte:SYNChronization:SEQUence:STATe?`

Returns 0 means no measurement signal is selected.

1 means a measurement signal is selected.

Examples `TRACE1:LTE:CONSTE:SYNCHRONIZATION:SEQUENCE:STATE?` might return 1, indicating that SSS or PSS is selected in the Trace tab of the LTE Constellation display.

TRACe<x>:NOISe:FIGure:AVERageCOUNt

Sets or queries the average counts for the specified trace in the Noise Figure display.

Conditions Measurement view: Noise Figure

The parameter <x>= 1 to 3.

Group Trace commands

Syntax
```
TRACe<x>:NOISe:FIGure:AVERageCOUNt <NR1>
TRACe<x>:NOISe:FIGure:AVERageCOUNt?
```

Arguments <NR1> = integer

Range: 2 to 10000

Returns <NR3> = floating point value with an exponent, which is the average count value.

Examples `TRAC1:NOIS:FIG:AVER:COUN?` might return 10.0000000000E+3, indicating that the average count value is set to 10000.

TRACe<x>:NOISe:FIGure:COUNt:ENABle

Sets or queries whether or not the count feature for the specified trace in the Noise Figure display is enabled or disabled.

Conditions Measurement view: Noise Figure display

The parameter <x>= 1 to 3.

Only available when trace Function is set to Avg (VRMS), Max Hold, or Min Hold.

Group Trace commands

Syntax
```
TRACe<x>:NOISe:FIGure:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe<x>:NOISe:FIGure:COUNt:ENABle?
```

Arguments OFF or 0 disables the count feature for the specified trace.

ON or 1 enables the count feature for the specified trace.

Returns 0 indicates the count feature for the specified trace if disabled.

1 indicates the count feature for the specified trace if enabled.

Examples `TRAC1:NOIS:FIG:COUN:ENAB ON` enables the count feature for Trace 1.

`TRAC1:NOIS:FIG:COUN:ENAB?` might return 0, indicating that the count feature for Trace 1 is disabled.

TRACe<x>:NOISe:FIGure:DETection

Sets or queries the trace detection method in the Noise Figure display.

Conditions Measurement view: Noise Figure display

The parameter <x> = 1 to 3.

Group Trace commands

Syntax
```
TRACe<x>:NOISe:FIGure:DETection { AVERage }
TRACe<x>:NOISe:FIGure:DETection?
```

Arguments `AVERage`: Each point on the trace is the result of determining the RMS Voltage value for all of the IQ samples available to the trace point. When displayed in either linear (Volts, Watts) or Log (dB, dBm), the correct RMS value results. When the averaging function is applied to a trace, the averaging is performed on the linear (Voltage) values, resulting in the correct average for RMS values.

Returns `AVER` means that the trace function is set to Average (VRMS).

Examples

`TRACE1:NOIS:FIG:DET AVER` sets the detection method for Trace 1 to Average (VRMS).

`TRACE2:NOIS:FIG:DET?` might return AVER, indicating that the function for Trace 1 is set to Average (VRMS).

TRACe<x>:NOISe:FIGure:FREeze

Specifies whether or not to freeze (halt updates of) the specified trace in the Noise Figure display.

Conditions

Measurement view: Noise Figure display

The parameter <x>= 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:FIGure:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:NOISe:FIGure:FREeze?
```

Arguments

OFF or 0 normally updates the display of the specified trace.

ON or 1 stops updating the display of the specified trace.

Returns

0 indicates the specified trace is being updated normally in the Noise Figure display.

1 indicates the specified trace is not being updated in the Noise Figure display.

Examples

`TRAC1:NOIS:FIG:FRE ON` freezes the display of Trace 1.

`TRAC1:NOIS:FIG:FRE?` might return 0, indicating that the display of Trace 1 is set to Freeze.

TRACe<x>:NOISe:FIgure:FUNCtion

Sets or queries the trace processing method (function) in the Noise Figure display.

Conditions

Measurement view: Noise Figure display

The parameter <x> = 1 to 3.

Group Trace commands

Syntax
```
TRACe<x>:NOISe:FIgure:FUNCtion { NORMal | AVERage | MAXHold | MINHold}
TRACe<x>:NOISe:FIgure:FUNCtion?
```

Arguments NORMal: Displays the trace record for each display point without additional processing.

AVERage: Each point on the trace is the result of determining the RMS Voltage value for all of the IQ samples available to the trace point. When displayed in either linear (Volts, Watts) or Log (dB, dBm), the correct RMS value results. When the averaging function is applied to a trace, the averaging is performed on the linear (Voltage) values, resulting in the correct average for RMS values.

MAXHold: Displays the maximum value in the trace record for each display point.

MINHold: Displays the minimum value in the trace record for each display point.

Returns NORM means that the trace function is set to Normal.

AVER means that the trace function is set to Average (VRMS).

MAXH means that the trace function is set to Max Hold.

MINH means that the trace function is set to Min Hold.

Examples TRAC1:NOIS:FIG:FUNC NORM sets the function for Trace 1 to Normal.

TRAC2:NOIS:FIG:FUNC? might return MAXH, indicating that the function for Trace 1 is set to Max Hold.

TRACe<x>:NOISe:FIGure:SELect

Sets the trace to be shown in the Noise Figure display. The query form of this command returns whether or not the specified trace is set to show (ON) or not (OFF).

Conditions Measurement view: Noise Figure display

The parameter <x> = 1 to 3.

Group Trace commands

Syntax

```
TRACe<x>:NOISe:FIGure:SELect
TRACe<x>:NOISe:FIGure:SELect?
```

Arguments

None

Returns

0 means that the specified trace is not shown (set to Off).

1 means that the specified trace is set to show (set to normal).

Examples

`TRAC1:NOIS:FIG:SEL` will set the display to show Trace 1.

`TRAC2:NOIS:FIG:SEL?` might return 1, indicating that Trace 2 is set to show.

TRACe<x>:NOISe:FIGure:SHOW

Sets or queries the state (shown or not shown) of the specified trace in the Noise Figure display.

Conditions

Measurement view: Noise Figure display

The parameter <x> = 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:FIGure:SHOW { OFF | ON | 0 | 1 }
TRACe<x>:NOISe:FIGure:SHOW?
```

Arguments

OFF or 0 hides the specified trace.

ON or 1 displays the specified trace.

Returns

0 indicates the specified trace is not shown in the Noise Figure display.

1 indicates the specified trace is shown in the Noise Figure display.

Examples

`TRAC1:NOIS:FIG:SHOW` sets Trace to show in the Noise Figure display.

`TRAC2:NOIS:FIG:SHOW?` might return 0, indicating that Trace 2 is not showing in the Noise Figure display.

TRACe<x>:NOISe:GAIN:AVERage:COUNt

Sets or queries the average counts for the specified trace in the Gain display.

Conditions Measurement view: Gain

The parameter <x>= 1 to 3.

Group Trace commands

Syntax

```
TRACe<x>:NOISe:GAIN:AVERage:COUNt <NR1>
TRACe<x>:NOISe:GAIN:AVERage:COUNt?
```

Arguments <NR1> = integer

Range: 2 to 10000

Returns <NR3> = floating point value with an exponent, which is the average count value.

Examples TRAC1:NOIS:GAIN:AVER:COUN? might return 10.0000000000E+3, indicating that the average count value is set to 10000.

TRACe<x>:NOISe:GAIN:COUNt:ENABle

Sets or queries whether or not the count feature for the specified trace in the Gain display is enabled or disabled.

Conditions Measurement view: Gain display

The parameter <x>= 1 to 3.

Only available when trace Function is set to Avg (VRMS), Max Hold, or Min Hold.

Group Trace commands

Syntax

```
TRACe<x>:NOISe:GAIN:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe<x>:NOISe:GAIN:COUNt:ENABle?
```

Arguments OFF or 0 disables the count feature for the specified trace.

ON or 1 enables the count feature for the specified trace.

Returns 0 indicates the count feature for the specified trace if disabled.

1 indicates the count feature for the specified trace if enabled.

Examples `TRAC1:NOIS:GAIN:COUN:ENAB ON` enables the count feature for Trace 1.

`TRAC1:NOIS:GAIN:COUN:ENAB?` might return 0, indicating that the count feature for Trace 1 is disabled.

TRACe<x>:NOISe:GAIN:DETection

Sets or queries the trace detection method in the Gain display.

Conditions Measurement view: Gain display

The parameter <x> = 1 to 3.

Group Trace commands

Syntax

```
TRACe<x>:NOISe:GAIN:DETection { AVERage }
TRACe<x>:NOISe:GAIN:DETection?
```

Arguments `AVERage`: Each point on the trace is the result of determining the RMS Voltage value for all of the IQ samples available to the trace point. When displayed in either linear (Volts, Watts) or Log (dB, dBm), the correct RMS value results. When the averaging function is applied to a trace, the averaging is performed on the linear (Voltage) values, resulting in the correct average for RMS values.

Returns `AVER` means that the trace function is set to Average (VRMS).

Examples `TRACE1:NOIS:YFAC:DET AVER` sets the detection method for Trace 1 to Average (VRMS).

`TRACE2:NOIS:YFAC:DET?` might return AVER, indicating that the function for Trace 1 is set to Average (VRMS).

TRACe<x>:NOISe:GAIN:FREeze

Specifies whether or not to freeze (halt updates of) the specified trace in the Gain display.

Conditions Measurement view: Gain display

The parameter <x>= 1 to 3.

Group Trace commands

Syntax

```
TRACe<x>:NOISe:GAIN:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:NOISe:GAIN:FREeze?
```

Arguments OFF or 0 normally updates the display of the specified trace.

ON or 1 stops updating the display of the specified trace.

Returns 0 indicates the specified trace is being updated normally in the Gain display.

1 indicates the specified trace is not being updated in the Gain display.

Examples `TRAC1:NOIS:GAIN:FRE ON` freezes the display of Trace 1.

`TRAC1:NOIS:GAIN:FRE?` might return 0, indicating that the display of Trace 1 is set to Freeze.

TRACe<x>:NOISe:GAIN:FUNCtion

Sets or queries the trace processing method (function) in the Gain display.

Conditions Measurement view: Gain display

The parameter <x> = 1 to 3.

Group Trace commands

Syntax

```
TRACe<x>:NOISe:GAIN:FUNCtion { NORMal | AVERage | MAXHold
| MINHold}
TRACe<x>:NOISe:GAIN:FUNCtion?
```

Arguments

NORMal: Displays the trace record for each display point without additional processing.

AVERage: Each point on the trace is the result of determining the RMS Voltage value for all of the IQ samples available to the trace point. When displayed in either linear (Volts, Watts) or Log (dB, dBm), the correct RMS value results. When the averaging function is applied to a trace, the averaging is performed on the linear (Voltage) values, resulting in the correct average for RMS values.

MAXHold: Displays the maximum value in the trace record for each display point.

MINHold: Displays the minimum value in the trace record for each display point.

Returns

NORM means that the trace function is set to Normal.

AVER means that the trace function is set to Average (VRMS).

MAXH means that the trace function is set to Max Hold.

MINH means that the trace function is set to Min Hold.

Examples

TRAC1:NOIS:GAIN:FUNC NORM sets the function for Trace 1 to Normal.

TRAC2:NOIS:GAIN:FUNC? might return MAXH, indicating that the function for Trace 1 is set to Max Hold.

TRACe<x>:NOISe:GAIN:SELect

Sets the trace to be shown in the Gain display. The query form of this command returns whether or not the specified trace is set to show (ON) or not (OFF).

Conditions

Measurement view: Gain display

The parameter <x> = 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:GAIN:SELect
TRACe<x>:NOISe:GAIN:SELect?
```

Arguments

None

Returns

0 means that the specified trace is not shown (set to Off).

1 means that the specified trace is set to show (set to normal).

Examples `TRAC1:NOIS:GAIN:SEL` will set the display to show Trace 1.

`TRAC2:NOIS:GAIN:SEL?` might return 1, indicating that Trace 2 is set to show.

TRACe<x>:NOISe:GAIN:SHOW

Sets or queries the state (shown or not shown) of the specified trace in the Gain display.

Conditions Measurement view: Gain display

The parameter <x> = 1 to 3.

Group Trace commands

Syntax
```
TRACe<x>:NOISe:GAIN:SHOW { OFF | ON | 0 | 1 }
TRACe<x>:NOISe:GAIN:SHOW?
```

Arguments OFF or 0 hides the specified trace.

ON or 1 displays the specified trace.

Returns 0 indicates the specified trace is not shown in the Gain display.

1 indicates the specified trace is shown in the Gain display.

Examples `TRAC1:NOIS:GAIN:SHOW` sets Trace to show in the Gain display.

`TRAC2:NOIS:GAIN:SHOW?` might return 0, indicating that Trace 2 is not showing in the Gain display.

TRACe<x>:NOISe:TEMPerature:AVERageCOUNt

Sets or queries the average counts for the specified trace in the Noise Temperature display.

Conditions Measurement view: Noise Temperature

The parameter <x>= 1 to 3.

Group Trace commands

Syntax

```
TRACe<x>:NOISe:TEMPerature:AVERageCOUNt <NR1>
TRACe<x>:NOISe:TEMPerature:AVERageCOUNt?
```

Arguments

<NR1> = integer

Range: 2 to 10000

Returns

<NR3> = floating point value with an exponent, which is the average count value.

Examples

`TRAC1:NOIS:TEMP:AVER:COUN?` might return 10.0000000000E+3, indicating that the average count value is set to 10000.

TRACe<x>:NOISe:TEMPerature:COUNt:ENABle

Sets or queries whether or not the count feature for the specified trace in the Noise Temperature display is enabled or disabled.

Conditions

Measurement view: Noise Temperature display

The parameter <x>= 1 to 3.

Only available when trace Function is set to Avg (VRMS), Max Hold, or Min Hold.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:TEMPerature:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe<x>:NOISe:TEMPerature:COUNt:ENABle?
```

Arguments

OFF or 0 disables the count feature for the specified trace.

ON or 1 enables the count feature for the specified trace.

Returns

0 indicates the count feature for the specified trace if disabled.

1 indicates the count feature for the specified trace if enabled.

Examples

`TRAC1:NOIS:TEMP:COUN:ENAB ON` enables the count feature for Trace 1.

`TRAC1:NOIS:TEMP:COUN:ENAB?` might return 0, indicating that the count feature for Trace 1 is disabled.

TRACe<x>:NOISe:TEMPerature:DETection

Sets or queries the trace detection method in the Noise Temperature display.

Conditions Measurement view: Noise Temperature display

The parameter <x> = 1 to 3.

Group Trace commands

Syntax

```
TRACe<x>:NOISe:TEMPerature:DETection { AVERage }
TRACe<x>:NOISe:TEMPerature:DETection?
```

Arguments AVERage: Each point on the trace is the result of determining the RMS Voltage value for all of the IQ samples available to the trace point. When displayed in either linear (Volts, Watts) or Log (dB, dBm), the correct RMS value results. When the averaging function is applied to a trace, the averaging is performed on the linear (Voltage) values, resulting in the correct average for RMS values.

Returns AVER means that the trace function is set to Average (VRMS).

Examples TRAC1:NOIS:TEMP:DET AVER sets the detection method for Trace 1 to Average (VRMS).

TRAC2:NOIS:TEMP:DET? might return AVER, indicating that the function for Trace 1 is set to Average (VRMS).

TRACe<x>:NOISE:TEMPerature:FREeze

Specifies whether or not to freeze (halt updates of) the specified trace in the Noise Temperature display.

Conditions Measurement view: Noise Temperature display

The parameter <x>= 1 to 3.

Group Trace commands

Syntax

```
TRACe<x>:NOISE:TEMPerature:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:NOISE:TEMPerature:FREeze?
```

Arguments OFF or 0 normally updates the display of the specified trace.

ON or 1 stops updating the display of the specified trace.

Returns 0 indicates the specified trace is being updated normally in the Noise Temperature display.

1 indicates the specified trace is not being updated in the Noise Temperature display.

Examples `TRAC1:NOIS:TEMP:FRE ON` freezes the display of Trace 1.

`TRAC1:NOIS:TEMP:FRE?` might return 0, indicating that the display of Trace 1 is set to Freeze.

TRACe<x>:NOISe:TEMPerature:FUNCtion

Sets or queries the trace processing method (function) in the Noise Temperature display.

Conditions Measurement view: Noise Temperature display

The parameter <x> = 1 to 3.

Group Trace commands

Syntax

```
TRACe<x>:NOISe:TEMPerature:FUNCtion { NORMal | AVERage |
MAXHold | MINHold}
TRACe<x>:NOISe:TEMPerature:FUNCtion?
```

Arguments `NORMal`: Displays the trace record for each display point without additional processing.

`AVERage`: Each point on the trace is the result of determining the RMS Voltage value for all of the IQ samples available to the trace point. When displayed in either linear (Volts, Watts) or Log (dB, dBm), the correct RMS value results. When the averaging function is applied to a trace, the averaging is performed on the linear (Voltage) values, resulting in the correct average for RMS values.

`MAXHold`: Displays the maximum value in the trace record for each display point.

`MINHold`: Displays the minimum value in the trace record for each display point.

Returns

`NORM` means that the trace function is set to Normal.

`AVER` means that the trace function is set to Average (VRMS).

`MAXH` means that the trace function is set to Max Hold.

`MINH` means that the trace function is set to Min Hold.

Examples

`TRAC1:NOIS:TEMP:FUNC NORM` sets the function for Trace 1 to Normal.

`TRAC2:NOIS:TEMP:FUNC?` might return MAXH, indicating that the function for Trace 1 is set to Max Hold.

TRACe<x>:NOISe:TEMPerature:SELect

Sets the trace to be shown in the Noise Temperature display. The query form of this command returns whether or not the specified trace is set to show (On) or not (Off).

Conditions

Measurement view: Noise Temperature display

The parameter <x> = 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:TEMPerature:SELect
TRACe<x>:NOISe:TEMPerature:SELect?
```

Arguments

None

Returns

0 means that the specified trace is not shown (set to Off).

1 means that the specified trace is set to show (set to normal).

Examples

`TRAC1:NOIS:TEMP:SEL` will set the display to show Trace 1.

`TRAC2:NOIS:TEMP:SEL?` might return 1, indicating that Trace 2 is set to show.

TRACe<x>:NOISe:TEMPerature:SHOW

Sets or queries the state (shown or not shown) of the specified trace in the Noise Temperature display.

Conditions Measurement view: Noise Temperature display

The parameter <x> = 1 to 3.

Group Trace commands

Syntax
```
TRACe<x>:NOISe:TEMPerature:SHOW { OFF | ON | 0 | 1 }
TRACe<x>:NOISe:TEMPerature:SHOW?
```

Arguments OFF or 0 hides the specified trace.

ON or 1 displays the specified trace.

Returns 0 indicates the specified trace is not shown in the Noise Temperature display.

1 indicates the specified trace is shown in the Noise Temperature display.

Examples `TRAC1:NOIS:TEMP:SHOW` sets Trace to show in the Noise Temperature display.

`TRAC2:NOIS:TEMP:SHOW?` might return 0, indicating that Trace 2 is not showing in the Noise Temperature display.

TRACe<x>:NOISe:YFACtor:AVERage:COUNt

Sets or queries the average counts for the specified trace in the Y Factor display.

Conditions Measurement view: Y Factor

The parameter <x>= 1 to 3.

Group Trace commands

Syntax
```
TRACe<x>:NOISe:YFACtor:AVERage:COUNt <NR1>
TRACe<x>:NOISe:YFACtor:AVERage:COUNt?
```

Arguments <NR1> = integer

Range: 2 to 10000

Returns <NR3> = floating point value with an exponent, which is the average count value.

Examples `TRAC1:NOIS:TEMP:AVER:COUN?` might return 10.0000000000E+3, indicating that the average count value is set to 10000.

TRACe<x>:NOISe:YFACtor:COUNt:ENABle

Sets or queries whether or not the count feature for the specified trace in the Y Factor display is enabled or disabled.

Conditions Measurement view: Y Factor display

The parameter <x>= 1 to 3.

Only available when trace Function is set to Avg (VRMS), Max Hold, or Min Hold.

Group Trace commands

Syntax
```
TRACe<x>:NOISe:YFACtor:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe<x>:NOISe:YFACtor:COUNt:ENABle?
```

Arguments OFF or 0 disables the count feature for the specified trace.

ON or 1 enables the count feature for the specified trace.

Returns 0 indicates the count feature for the specified trace if disabled.

1 indicates the count feature for the specified trace if enabled.

Examples `TRAC1:NOIS:YFAC:COUN:ENAB ON` enables the count feature for Trace 1.

`TRAC1:NOIS:YFAC:COUN:ENAB?` might return 0, indicating that the count feature for Trace 1 is disabled.

TRACe<x>:NOISe:YFACtor:DETection

Sets or queries the trace detection method in the Y Factor display.

Conditions Measurement view: Y Factor display

The parameter <x> = 1 to 3.

Group Trace commands

Syntax

```
TRACe<x>:NOISe:YFACtor:DETection { AVERage }
TRACe<x>:NOISe:YFACtor:DETection?
```

Arguments

AVERage: Each point on the trace is the result of determining the RMS Voltage value for all of the IQ samples available to the trace point. When displayed in either linear (Volts, Watts) or Log (dB, dBm), the correct RMS value results. When the averaging function is applied to a trace, the averaging is performed on the linear (Voltage) values, resulting in the correct average for RMS values.

Returns

AVER means that the trace function is set to Average (VRMS).

Examples

TRACE1:NOIS:YFAC:DET AVER sets the detection method for Trace 1 to Average (VRMS).

TRACE2:NOIS:YFAC:DET? might return AVER, indicating that the function for Trace 1 is set to Average (VRMS).

TRACe<x>:NOISE:YFACtor:FREeze

Specifies whether or not to freeze (halt updates of) the specified trace in the Y Factor display.

Conditions

Measurement view: Y Factor display

The parameter <x>= 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISE:YFACtor:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:NOISE:YFACtor:FREeze?
```

Arguments

OFF or 0 normally updates the display of the specified trace.

ON or 1 stops updating the display of the specified trace.

Returns

0 indicates the specified trace is being updated normally in the Y Factor display.

1 indicates the specified trace is not being updated in the Y Factor display.

Examples

`TRAC1:NOIS:YFAC:FRE ON` freezes the display of Trace 1.

`TRAC1:NOIS:YFAC:FRE?` might return 0, indicating that the display of Trace 1 is set to Freeze.

TRACe<x>:NOISe:YFACtor:FUNCtion

Sets or queries the trace processing method (function) in the Y Factor display.

Conditions

Measurement view: Y Factor display

The parameter <x> = 1 to 3.

Group

Trace commands

Syntax

```
TRACe<x>:NOISe:YFACtor:FUNCtion { NORMal | AVERage | MAXHold
| MINHold}
TRACe<x>:NOISe:YFACtor:FUNCtion?
```

Arguments

`NORMal`: Displays the trace record for each display point without additional processing.

`AVERage`: Each point on the trace is the result of determining the RMS Voltage value for all of the IQ samples available to the trace point. When displayed in either linear (Volts, Watts) or Log (dB, dBm), the correct RMS value results. When the averaging function is applied to a trace, the averaging is performed on the linear (Voltage) values, resulting in the correct average for RMS values.

`MAXHold`: Displays the maximum value in the trace record for each display point.

`MINHold`: Displays the minimum value in the trace record for each display point.

Returns

`NORM` means that the trace function is set to Normal.

`AVER` means that the trace function is set to Average (VRMS).

`MAXH` means that the trace function is set to Max Hold.

`MINH` means that the trace function is set to Min Hold.

Examples

`TRAC1:NOIS:YFAC:FUNC NORM` sets the function for Trace 1 to Normal.

`TRAC2:NOIS:YFAC:FUNC?` might return MAXH, indicating that the function for Trace 1 is set to Max Hold.

TRACe<x>:NOISe:YFACtor:SELect

Sets the trace to be shown in the Y Factor display. The query form of this command returns whether or not the specified trace is set to show (ON) or not (OFF).

Conditions Measurement view: Y Factor display

The parameter <x> = 1 to 3.

Group Trace commands

Syntax

```
TRACe<x>:NOISe:YFACtor:SELect
TRACe<x>:NOISe:YFACtor:SELect?
```

Arguments None

Returns 0 means that the specified trace is not shown (set to Off).

1 means that the specified trace is set to show (set to normal).

Examples `TRAC1:NOIS:YFAC:SEL` will set the display to show Trace 1.

`TRAC2:NOIS:YFAC:SEL?` might return 1, indicating that Trace 2 is set to show.

TRACe<x>:NOISe:YFACtor:SHOW

Sets or queries the state (shown or not shown) of the specified trace in the Y Factor display.

Conditions Measurement view: Y Factor display

The parameter <x> = 1 to 3.

Group Trace commands

Syntax

```
TRACe<x>:NOISe:YFACtor:SHOW { OFF | ON | 0 | 1 }
TRACe<x>:NOISe:YFACtor:SHOW?
```

Arguments OFF or 0 hides the specified trace.

ON or 1 displays the specified trace.

Returns 0 indicates the specified trace is not shown in the Y Factor display.

1 indicates the specified trace is shown in the Y Factor display.

Examples `TRACE1:NOISE:YFACTOR:SHOW` sets Trace to show in the Y Factor display.

`TRACE2:NOISE:YFACTOR:SHOW?` might return 0, indicating that Trace 2 is not showing in the Y Factor display.

TRACe<x>:OFDM:EVM

Determines whether or not to show the specified trace in the OFDM EVM view.

The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions Measurement views: OFDM

Group Trace commands

Syntax
```
TRACe<x>:OFDM:EVM { OFF | ON | 0 | 1 }
TRACe<x>:OFDM:EVM?
```

Arguments `OFF` or `0` hides the specified trace in the view.

`ON` or `1` shows the specified trace in the view.

Examples `TRACe1:OFDM:EVM ON` shows Trace 1 in the OFDM EVM view.

TRACe<x>:OFDM:EVM:FREeze

Determines whether or not to freeze the display of the specified trace in the OFDM EVM view.

The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions Measurement views: OFDM

Group Trace commands

Syntax
```
TRACe<x>:OFDM:EVM:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:OFDM:EVM:FREeze?
```

Arguments OFF or 0 updates the display of the specified trace normally.

ON or 1 stops updating the display of the specified trace.

Examples TRACE1:OFDM:EVM:FREEZE ON freezes the display for Trace 1.

TRACe<x>:OFDM:EVM:SELect

Sets or queries the specified trace to display in the OFDM EVM view.

The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions Measurement views: OFDM

Group Trace commands

Syntax
```
TRACe<x>:OFDM:EVM:SELect
TRACe<x>:OFDM:EVM:SELect?
```

Arguments None

Returns { OFF | ON | 0 | 1 }

OFF or 0 indicates that the readout is not being displayed for the specified trace.

ON or 1 indicates that the readout is being displayed for the specified trace.

Examples TRACE1:OFDM:EVM:SELECT selects Trace 1 (Average trace) to display the readout.

TRACE1:OFDM:EVM:SELECT? might return 1, indicating that the readout is being displayed for Trace 1.

TRACe<x>:OFDM:MERRor

Determines whether or not to show the specified trace in the OFDM Magnitude error view.

The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions Measurement views: OFDM

Group Trace commands

Syntax
```
TRACe<x>:OFDM:MERRor { OFF | ON | 0 | 1 }
TRACe<x>:OFDM:MERRor?
```

Arguments `OFF` or `0` hides the specified trace in the view.

`ON` or `1` shows the specified trace in the view.

Examples `TRACe1:OFDM:MERRor ON` shows Trace 1 in the OFDM Magnitude error view.

TRACe<x>:OFDM:MERRor:FREeze

Determines whether or not to freeze the display of the specified trace in the OFDM Magnitude error view.

The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions Measurement views: OFDM

Group Trace commands

Syntax
```
TRACe<x>:OFDM:MERRor:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:OFDM:MERRor:FREeze?
```

Arguments `OFF` or `0` updates the display of the specified trace normally.

`ON` or `1` stops updating the display of the specified trace.

Examples `TRACE1:OFDM:MERRor:FREEZE ON` freezes the display for Trace 1.

TRACe<x>:OFDM:MERRor:SELect

Sets or queries the specified trace to display in the OFDM Magnitude error view.

The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions Measurement views: OFDM

Group Trace commands

Syntax

```
TRACe<x>:OFDM:MERRor:SELect { OFF | ON | 0 | 1 }
TRACe<x>:OFDM:MERRor:SELect?
```

Arguments None

Returns { OFF | ON | 0 | 1 }

`OFF` or `0` indicates that the readout is not being displayed for the specified trace.

`ON` or `1` indicates that the readout is being displayed for the specified trace.

Examples `TRACE1:OFDM:MERRor:SELECT` selects Trace 1 (Average trace) to display the readout.

`TRACE1:OFDM:MERRor:SELECT?` might return 1, indicating that the readout is being displayed for Trace 1.

TRACe<x>:OFDM:PERRor

Determines whether or not to show the specified trace in the OFDM Phase error view.

The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions Measurement views: OFDM

Group Trace commands

Syntax

```
TRACe<x>:OFDM:PERRor { OFF | ON | 0 | 1 }
TRACe<x>:OFDM:PERRor?
```

Arguments `OFF` or `0` hides the specified trace in the view.

`ON` or `1` shows the specified trace in the view.

Examples `TRACe1:OFDM:PERRor ON` shows Trace 1 in the OFDM Phase error view.

TRACe<x>:OFDM:PERRor:FREeze

Determines whether or not to freeze the display of the specified trace in the OFDM Phase error view.

The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions Measurement views: OFDM

Group Trace commands

Syntax
```
TRACe<x>:OFDM:PERRor:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:OFDM:PERRor:FREeze?
```

Arguments OFF or 0 updates the display of the specified trace normally.

ON or 1 stops updating the display of the specified trace.

Examples TRACE1:OFDM:PERRor:FREEZE ON freezes the display for Trace 1.

TRACe<x>:OFDM:PERRor:SELect

Sets or queries the specified trace to display in the OFDM Phase error view.

The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions Measurement views: OFDM

Group Trace commands

Syntax
```
TRACe<x>:OFDM:PERRor:SELect { OFF | ON | 0 | 1 }
TRACe<x>:OFDM:PERRor:SELect?
```

Arguments None

Returns { OFF | ON | 0 | 1 }

OFF or 0 indicates that the readout is not being displayed for the specified trace.

ON or 1 indicates that the readout is being displayed for the specified trace.

Examples TRACE1:OFDM:PERRor:SELECT selects Trace 1 (Average trace) to display the readout.

TRACE1:OFDM:PERRor:SELECT? might return 1, indicating that the readout is being displayed for Trace 1.

TRACe<x>:OFDM:POWer

Determines whether or not to show the specified trace in the OFDM Power view.

The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions Measurement views: OFDM

Group Trace commands

Syntax
```
TRACe<x>:OFDM:POWer { OFF | ON | 0 | 1 }
TRACe<x>:OFDM:POWer?
```

Arguments OFF or 0 hides the specified trace in the view.

ON or 1 shows the specified trace in the view.

Examples TRACe1:OFDM:POWer ON shows Trace 1 in the OFDM Power view.

TRACe<x>:OFDM:POWer:FREeze

Determines whether or not to freeze the display of the specified trace in the OFDM Power view.

The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions Measurement views: OFDM

Group Trace commands

Syntax
```
TRACe<x>:OFDM:POWer:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:OFDM:POWer:FREeze?
```

Arguments OFF or 0 updates the display of the specified trace normally.

ON or 1 stops updating the display of the specified trace.

Examples `TRACE1:OFDM:POWer:FREEZE ON` freezes the display for Trace 1.

TRACe<x>:OFDM:POWer:SELect

Sets or queries the specified trace to display in the OFDM Power view.

The parameter <x> = 1 or 2, where 1 is Average and 2 is Value.

Conditions Measurement views: OFDM

Group Trace commands

Syntax
```
TRACe<x>:OFDM:POWer:SELect { OFF | ON | 0 | 1 }
TRACe<x>:OFDM:POWer:SELect?
```

Arguments None

Returns { OFF | ON | 0 | 1 }

`OFF` or `0` indicates that the readout is not being displayed for the specified trace.

`ON` or `1` indicates that the readout is being displayed for the specified trace.

Examples `TRACE1:OFDM:POWer:SELECT` selects Trace 1 (Average trace) to display the readout.

`TRACE1:OFDM:POWer:SELECT?` might return 1, indicating that the readout is being displayed for Trace 1.

TRACe<x>:P25:CONStellation

Specifies whether or not to show the specified trace (Trace 1 or 2) in the P25 Constellation display.

Conditions Measurement view: P25 Constellation display

The parameter <x>= 1 or 2.

Group Trace commands

Syntax

```
TRACe<x>:P25:CONStellation { OFF | ON | 0 | 1 }
TRACe<x>:P25:CONStellation?
```

Arguments

OFF or 0 hides the specified trace.

ON or 1 displays the specified trace.

Returns

0 indicates the specified trace is not shown in the P25 Constellation display.

1 indicates the specified trace is shown in the P25 Constellation display.

Examples

`TRAC1:P25:CONS ON` specifies that Trace 1 be shown in the P25 Constellation display.

TRACe<x>:P25:CONStellation:FREeze

Specifies whether or not to freeze (halt updates of) the specified trace (Trace 1 or 2) in the P25 Constellation display.

Conditions

Measurement view: P25 Constellation display

The parameter <x>= 1 or 2.

Group

Trace commands

Syntax

```
TRACe<x>:P25:CONStellation:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:P25:CONStellation:FREeze?
```

Arguments

OFF or 0 normally updates the display of the specified trace.

ON or 1 stops updating the display of the specified trace.

Returns

0 indicates the specified trace is being updated normally in the P25 Constellation display.

1 indicates the specified trace is not being updated in the P25 Constellation display.

Examples

`TRAC1:P25:CONS:FRE ON` freezes the display of Trace 1.

`TRAC1:P25:CONS:FRE?` might return 0, indicating that the display of Trace 1 is set to Freeze.

TRACe<x>:P25:CONStellation:MODE

Sets or queries how to display the trace content in the P25 Constellation display.

Conditions Measurement view: P25 Constellation display

The parameter <x>= 1 or 2.

Group Trace commands

Syntax

```
TRACe<x>:P25:CONStellation:MODE { VECT | SYMB | LIN }
```

Arguments VECT (vectors) connects adjacent symbol points with the signal locus.

SYMB (symbols) displays individual symbol points.

LIN (line) displays all symbol points as a line.

Returns VECT: adjacent symbol points are connected with the signal locus.

SYMB: individual symbol points are displayed.

LIN: all symbol points are displayed as a line.

Examples `TRAC1:P25:CONS:MODE SYMB` sets the display to show the constellation of Trace 1 connecting adjacent symbol points with the signal locus.

TRACe<x>:P25:CONStellation:SELect

Selects the specified trace (Trace 1 or 2) in the P25 Constellation display.

Conditions Measurement view: P25 Constellation display

The parameter <x>= 1 or 2.

Group Trace commands

Syntax

```
TRACe<x>:P25:CONStellation:SELect { OFF | ON | 0 | 1 }
TRACe<x>:P25:CONStellation:SELect?
```

Arguments None

Returns 0 indicates the specified trace is not the active trace in the P25 Constellation display.

1 indicates the specified trace is the active trace in the P25 Constellation display.

Examples `TRAC1:P25:CONS:SEL?` might return 1, indicating that Trace 1 is the selected trace in the P25 Constellation display.

TRACe<x>:P25:EDIagram:ENABle:I

Determines whether to show or hide the I trace in the P25 Eye Diagram display.

Conditions Measurement view: P25 Eye Diagram

Group Trace commands

Syntax
```
TRACe<x>:P25:EDIagram:ENABle:I
TRACe<x>:P25:EDIagram:ENABle:I?
```

Related Commands TRACe<x>:P25:EDIagram:ENABle:Q

Arguments OFF or 0 hides the I trace.

ON or 1 shows the I trace.

Returns 0 indicates the trace is off.

1 indicates the trace is on.

Examples `TRACEX:P25:EDIAGRAM:ENABLE:I ON` shows the I trace in the P25 Eye Diagram display.

TRACe<x>:P25:EDIagram:ENABle:Q

Determines whether to show or hide the Q trace in the P25 Eye Diagram display.

Conditions Measurement view: P25 Eye Diagram

Group Trace commands

Syntax

```
TRACe<x>:P25:EDIagram:ENABle:Q
TRACe<x>:P25:EDIagram:ENABle:Q?
```

Related Commands TRACe<x>:P25:EDIagram:ENABle:I

Arguments OFF or 0 hides the Q trace.

ON or 1 shows the Q trace.

Returns 0 indicates the trace is off.

1 indicates the trace is on.

Examples `TRACEX:P25:EDIAGRAM:ENABLE:Q ON` shows the Q trace in the P25 Eye Diagram display.

TRACe<x>:P25:EDIagram:SELect:I

This command selects the I trace in the P25 Eye Diagram display. The query version of this command returns whether the I trace is selected or not.

Conditions Measurement view: P25 Eye Diagram

Group Trace commands

Syntax

```
TRACe<x>:P25:EDIagram:SELect:I
TRACe<x>:P25:EDIagram:SELect:I?
```

Related Commands TRACe<x>:P25:EDIagram:SELect:Q

Arguments None.

Returns 0 indicates that the I trace is not the selected trace.

1 indicates that the I trace is the selected trace.

Examples `TRACEX:P25:EDIAGRAM:SELECT:I` selects the I trace in the P25 Eye Diagram display.

TRACe<x>:P25:EDIagram:SELect:Q

This command selects the Q trace in the P25 Eye Diagram display. The query version of this command returns whether the Q trace is selected or not.

Conditions Measurement view: P25 Eye Diagram

Group Trace commands

Syntax

```
TRACe<x>:P25:EDIagram:SELect:Q
TRACe<x>:P25:EDIagram:SELect:Q?
```

Related Commands TRACe<x>:P25:EDIagram:SELect:I

Arguments None.

Returns 0 indicates that the Q trace is not the selected trace.

1 indicates that the Q trace is the selected trace.

Examples `TRACEX:P25:EDIAGRAM:SELECT:Q` selects the Q trace in the P25 Eye Diagram display.

TRACe<x>:PHVTime:DETection

Sets or queries the display detector (method to be used for decimating traces to fit the available horizontal space on screen). The number of horizontal pixels on screen is generally smaller than that of waveform data points. When actually displayed, the waveform data is therefore thinned out, according to the number of pixels, for being compressed.

Conditions Measurement views: Phase versus Time

Group Trace commands

Syntax

```
TRACe<x>:PHVTime:DETection { AVERage | POSitive | NEGative |
POSNegative | SAMPle }
TRACe<x>:PHVTime:DETection?
```

Arguments

AVERage displays the average data value for each pixel.

POSitive displays the maximum data value for each pixel.

NEGative displays the minimum data value for each pixel.

POSNegative displays the maximum and minimum data values for each pixel.

SAMPle displays the first sample value received for each pixel.

Examples

TRACe1:PHVTime:DETection POSitive displays the maximum data value for each pixel.

TRACe<x>:PNOise:SELect

Selects the trace in the phase noise measurement. The query returns the currently selected trace.

The parameter <x> = 1 and 2.

Conditions

Measurement views: Phase noise

Group

Trace commands

Syntax

```
TRACe<x>:PNOise:SELect
TRACe<x>:PNOise:SELect?
```

Arguments

None

Returns

0 (not selected) or 1 (selected).

Examples

TRACE2:PNOISE:SELECT selects Trace 2.

TRACe<x>:PNOise:SHOW

Shows or hides the specified trace in the phase noise view.

The parameter <x> = 1 and 2.

Conditions

Measurement views: Phase noise

Group Trace commands

Syntax
```
TRACe<x>:PNOise:SHOW { OFF | ON | 0 | 1 }
TRACe<x>:PNOise:SHOW?
```

Arguments `OFF` or `0` hides the specified trace.

`ON` or `1` shows the specified trace.

Examples `TRACE1:PNOISE:SHOW ON` shows Trace 1 in the phase noise view.

TRACe<x>:PNOise:SMOothing:COUNt

Sets or queries the number of data points to take the moving average for smoothing the trace. This command is effective when TRACe<x>:PNOise:SMOothing:ENABle is set to ON.

The parameter <x> = 1 and 2.

Conditions Measurement views: Phase noise

Group Trace commands

Syntax
```
TRACe<x>:PNOise:SMOothing:COUNt <number>
TRACe<x>:PNOise:SMOothing:COUNt?
```

Arguments `<number> ::= <NR1>` specifies the number of data points to take the moving average for smoothing. Range: 3 to 50.

Examples `TRACE1:PNOISE:SMOOTHING:COUNT 16` sets the smoothing count to 16 for Trace 1.

TRACe<x>:PNOise:SMOothing:ENABle

Determines whether to enable or disable smoothing the specified trace in the phase noise view.

The parameter <x> = 1 and 2.

Conditions Measurement views: Phase noise

Group Trace commands

Syntax
```
TRACe<x>:PNOise:SMOothing:ENABle { OFF | ON | 0 | 1 }
TRACe<x>:PNOise:SMOothing:ENABle?
```

Arguments `OFF` or `0` disables smoothing.

`ON` or `1` enables smoothing.

Examples `TRACE1:PNOISE:SMOOTHING:ENABLE ON` enables smoothing Trace 1 in the phase noise view.

TRACe<x>:PNOise:SMOothing:RESet (No Query Form)

Restarts the smoothing process, discarding accumulated data and resetting the counter. This command is effective when TRACe<x>:PNOise:SMOothing:ENABle is set to ON.

The parameter <x> = 1 and 2.

Conditions Measurement views: Phase noise

Group Trace commands

Syntax
```
TRACe<x>:PNOise:SMOothing:RESet
```

Arguments None

Examples `TRACE1:PNOISE:SMOOTHING:RESET` restarts the smoothing process for Trace 1.

TRACe<x>:SPECtrum

Determines whether to show or hide the specified trace in the Spectrum view.

The parameter <x> = 1 to 5; All traces are valid.

Conditions Measurement views: Spectrum

Group Trace commands

Syntax

```
TRACe<x>:SPECtrum { OFF | ON | 0 | 1 }
TRACe<x>:SPECtrum?
```

Arguments

OFF or 0 hides the specified trace.

ON or 1 shows the specified trace.

For Trace 5 (spectrogram), use the TRACe:SGRam:SELect:LINE command to select the line to send to the spectrum display.

Examples

TRACE1:SPECTRUM ON shows Trace 1 in the Spectrum Analyzer view.

TRACe<x>:SPECtrum:AVERage:COUNt

Sets or queries the number of traces to combine. This command is effective when TRACe<x>:SPECtrum:FUNCtion is AVERage, MAXHold or MINHold.

The parameter <x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

Conditions

Measurement views: Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:SPECtrum:AVERage:COUNt <number>
TRACe<x>:SPECtrum:AVERage:COUNt?
```

Arguments

<number> :: <NR1> specifies the number of traces to combine for averaging. Range: 1 to 10000.

Examples

TRACE1:SPECTRUM:AVERAGE:COUNT 64 sets the average count to 64 for Trace 1.

TRACe<x>:SPECtrum:AVERage:RESet (No Query Form)

Clears average data and counter, and restarts the average process for the specified trace in the Spectrum view. This command is effective when TRACe<x>:SPECtrum:FUNCtion is set to AVERage, MAXHold or MINHold.

The parameter <x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

Conditions Measurement views: Spectrum

Group Trace commands

Syntax `TRACe<x>:SPECtrum:AVERage:RESet`

Arguments None

Examples `TRACE1:SPECTRUM:AVERAGE:RESET` clears average data and counter, and restarts the average process for Trace 1.

TRACe<x>:SPECtrum:COUNt

Sets or queries how many acquisitions run in the single acquisition mode for the Max or Min Hold trace in the Spectrum measurement. This command is effective when TRACe<x>:SPECtrum:FUNCtion is set to MAXHold or MINHold and INITiate:CONTinuous is set to OFF.

The parameter <x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

Conditions Measurement views: Spectrum

Group Trace commands

Syntax
```
TRACe<x>:SPECtrum:COUNt <number>
TRACe<x>:SPECtrum:COUNt?
```

Arguments `<number> ::= <NR1>` specifies the count for Max/Min Hold. Range: 1 to 10000.

Examples `TRACE1:SPECTRUM:COUNT 32` sets the count to 32 for Trace 1.

TRACe<x>:SPECtrum:COUNt:ENABle

Determines whether to enable or disable the count for the Max or Min Hold trace in the Spectrum view. This command is effective when TRACe<x>:SPECtrum:FUNCtion is set to MAXHold or MINHold.

The parameter <x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

Conditions Measurement views: Spectrum

Group Trace commands

Syntax
```
TRACe<x>:SPECtrum:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe<x>:SPECtrum:COUNt:ENABle?
```

Arguments `OFF` or `0` disables the count for the Max/Min Hold trace.

`ON` or `1` enables the count for the Max/Min Hold trace.

Examples `TRACE1:SPECTRUM:COUNT:ENABLE ON` enables the Max/Min Hold count for Trace 1.

TRACe<x>:SPECtrum:COUNt:RESet (No Query Form)

Clears the Max or Min Hold data and counter, and restarts the process for the specified trace in the Spectrum view. This command is effective when TRACe<x>:SPECtrum:FUNCtion is set to MAXHold or MINHold.

The parameter <x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

Conditions Measurement views: Spectrum

Group Trace commands

Syntax
```
TRACe<x>:SPECtrum:COUNt:RESet
```

Arguments None

Examples `TRACE1:SPECTRUM:COUNT:RESET` clears the Max/Min Hold data and counter, and restarts the process for Trace 1.

TRACe<x>:SPECtrum:DETection

Sets or queries the display detector (method to be used for decimating traces to fit the available horizontal space on screen). The number of horizontal pixels on screen is generally smaller than that of waveform data points. When actually displayed, the waveform data is therefore thinned out, according to the number of pixels, for being compressed.

The parameter <x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

Conditions Measurement views: Spectrum

Group Trace commands

Syntax

```
TRACe<x>:SPECtrum:DETection { AVERage | POSitive | NEGative
| CAVerage | CPEak | QUASipeak | SAMPle | AVGLog }
TRACe<x>:SPECtrum:DETection?
```

Arguments

`AVERage` displays the average data value for each pixel.

`POSitive` displays the maximum data value for each pixel.

`NEGative` displays the minimum data value for each pixel.

`CAVerage` displays the CISPR average value for each pixel.

`CPEak` displays the CISPR peak value for each pixel.

`QUASipeak` displays the quasi-peak value for each pixel.

`SAMPle` displays the most recent sample value for each pixel.

`AVGLog` displays the average data value of logs for each pixel.

Examples `TRACE1:SPECTRUM:DETECTION POSitive` displays the maximum data value for each pixel on Trace 1.

TRACe<x>:SPECtrum:FREeze

Determines whether or not to freeze the display of the specified trace in the Spectrum view.

The parameter <x> = 1 to 5; All traces are valid.

Conditions Measurement views: Spectrum

Group Trace commands

Syntax
```
TRACe<x>:SPECtrum:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:SPECtrum:FREeze?
```

Arguments `OFF` or `0` updates the display of the specified trace normally.

`ON` or `1` stops updating the display of the specified trace.

Examples `TRACE1:SPECTRUM:FREEZE ON` freezes Trace 1 in the Spectrum Analysis display.

TRACe<x>:SPECtrum:FUNCtion

Sets or queries the function for the specified trace in the Spectrum view.

The parameter <x> = 1 to 3; Trace 4 (math trace) and Trace 5 (spectrogram) are invalid.

Conditions Measurement views: Spectrum

Group Trace commands

Syntax
```
TRACe<x>:SPECtrum:FUNCtion { NONE | AVERage | MAXHold |
MINHold | AVGLog}
TRACe<x>:SPECtrum:FUNCtion?
```

Arguments `NONE` selects the normal spectrum display.

`AVERage` selects the Average display that indicates the average signal level at each frequency point.

`MAXHold` selects the Max Hold display that indicates the maximum signal level at each frequency point.

`MINHold` selects the Min Hold display that indicates the minimum signal level at each frequency point.

`AVGLog` selects the Average of logs display that indicates the average signal level of logs at each frequency point.

Examples `TRACE1:SPECTRUM:FUNCTION MAXHold` selects Max Hold for Trace 1 in the Spectrum view.

TRACe<x>:SPECtrum:LEFToperand

Sets or queries the left operand for the math trace (Trace 4) in the Spectrum view.

The parameter <x> = 4; Only Trace 4 (math trace) is valid.

Conditions Measurement views: Spectrum

Group Trace commands

Syntax

```
TRACe<x>:SPECtrum:LEFToperand { TRACE1 | TRACE2 | TRACE3 }
TRACe<x>:SPECtrum:LEFToperand?
```

Related Commands TRACe<x>:SPECtrum:RIGHtoperand

Arguments `TRACE1` selects Trace 1 as the left operand for the math trace.

`TRACE2` selects Trace 2 as the left operand for the math trace.

`TRACE3` selects Trace 3 as the left operand for the math trace.

Examples `TRACE4:SPECTRUM:LEFTOPERAND TRACE1` selects Trace 1 as the left operand for the math trace.

TRACe<x>:SPECtrum:RIGHtoperand

Sets or queries the right operand for the math trace (Trace 4) in the Spectrum view.

The parameter <x> = 4; Only Trace 4 (math trace) is valid.

Conditions Measurement views: Spectrum

Group Trace commands

Syntax

```
TRACe<x>:SPECtrum:RIGHtoperand { TRACE1 | TRACE2 | TRACE3 }
TRACe<x>:SPECtrum:RIGHtoperand?
```

Related Commands TRACe<x>:SPECtrum:LEFToperand

Arguments

TRACE1 selects Trace 1 as the right operand for the math trace.

TRACE2 selects Trace 2 as the right operand for the math trace.

TRACE3 selects Trace 3 as the right operand for the math trace.

Examples

TRACE4:SPECTRUM:RIGHTOPERAND TRACE1 selects Trace 1 as the right operand for the math trace.

TRACe<x>:SPECtrum:SELect

Sets or queries the trace to display the readout at the upper left of the Spectrum view.

The parameter <x> = 1 to 5; All traces are valid.

NOTE. *TRACe5 (spectrogram) is valid when the spectrum and spectrogram measurements are running.*

Conditions

Measurement views: Spectrum

Group

Trace commands

Syntax

```
TRACe<x>:SPECtrum:SELect
TRACe<x>:SPECtrum:SELect?
```

Arguments

None

Returns

{ 0 | 1 }

0 indicates that the readout is not being displayed for the specified trace.

1 indicates that the readout is being displayed for the specified trace.

Examples

TRACE1:SPECTRUM:SELECT selects Trace 1 to display the readout.

TRACE1:SPECTRUM:SELECT? might return 1, indicating that the readout is being displayed for Trace 1.

TRACe<x>:TXGain

Sets or queries the display to show or hide the specified trace.

Conditions Measurement view: Transmission Gain

Where <x> is the trace and is 1, 2, or 3.

Group Display group

Syntax
```
TRACe<x>:TXGain { OFF | ON | 0 | 1 }
TRACe<x>:TXGain?
```

Arguments OFF or 0 hides the specified trace.

ON or 1 shows the specified trace.

Returns OFF or 0 means that the specified trace is hidden.

ON or 1 means that the specified trace is showing.

Examples `TRACE1:TXGAIN 1` shows Trace 1.

TRACe<x>:TXGain:AVERage:COUNt

Sets or queries the number of traces averaged to generate the displayed trace. (Present only when Function is set to anything but Normal.)

Conditions Measurement view: Transmission Gain

Where <x> is the trace and is 1, 2, or 3.

Group Display group

Syntax `TRACe<x>:TXGain:AVERage:COUNt <number>`

Arguments <number>::=<NR1> specifies the number of traces to combine for averaging.

Range: 1 to 10000

Returns <NR1> is an integer that is the number of traces to combine for averaging.

Examples `TRACE1:TXGAIN:AVERAGE:COUNT 64` sets the average count to 64 for Trace 1.

TRACe<x>:TXGain:AVERage:RESet (No Query Form)

Clears/resets the average data and counter and restarts the averaging process.

Conditions Measurement view: Transmission Gain

Where <x> is the trace and is 1, 2, or 3.

Function must be is set to anything but Normal.

Group Display group

Syntax `TRACe<x>:TXGain:AVERage:RESet`

Arguments None.

Examples `TRACE1:TXGAIN:AVERAGE:RESET` clears average data and counter, and restarts the average process for Trace 1.

TRACe<x>:TXGain:COUNt

Sets or queries how many acquisitions run in the single acquisition mode for the Average or Max or Min Hold trace in the measurement.

Conditions Measurement view: Transmission Gain

Where <x> is the trace and is 1, 2, or 3.

Function must be is set to anything but Normal.

Group Display group

Syntax
```
TRACe<x>:TXGain:COUNt <number>
TRACe<x>:TXGain:COUNt?
```

Arguments <number>::=<NR1> specifies the count for Averaged, Max or Min Hold.

Range: 1 to 10000.

Returns <NR1> is an integer that is the count for Averaged, Max or Min Hold.

Examples `TRACE1:TXGAIN:COUNT 32` sets the count to 32 for Trace 1.

TRACe<x>:TXGain:COUNt:ENABle

The command enables or disables whether to use the number of traces Averaged or Max Hold or Min Hold to generate the displayed trace. The query returns whether or not it is enabled.

Conditions Measurement view: Transmission Gain

Where <x> is the trace and is 1, 2, or 3.

Group Display group

Syntax
```
TRACe<x>:TXGain:COUNt:ENABle { OFF | ON | 0 | 1 }
TRACe<x>:TXGain:COUNt:ENABle?
```

Arguments OFF or 0 disables the count for the specified trace.

ON or 1 enables the count for the specified trace.

Returns OFF or 0 means that the count for the specified trace is disabled.

ON or 1 means that the count for the specified trace is enabled.

Examples `TRACE1:TXGAIN:COUNT:ENABLE ON` enables the count for Trace 1.

TRACe<x>:TXGain:COUNt:RESet (No Query Form)

Restarts the trace count when the function is Average, Max Hold, or Min Hold.

Conditions Measurement view: Transmission Gain

Where <x> is the trace and is 1, 2, or 3.

Group Display group

Syntax
```
TRACe<x>:TXGain:COUNt:RESet
```

Arguments None.

Examples `TRACE1:TXGAIN:COUNT:RESET` clears the trace data and counter, and restarts the process for Trace 1.

TRACe<x>:TXGain:FREeze

Sets the trace as frozen or live. The query returns whether the trace is frozen or live.

Conditions Measurement view: Transmission Gain

Where <x> is the trace and is 1, 2, or 3.

Group Display group

Syntax

```
TRACe<x>:TXGain:FREeze { OFF | ON | 0 | 1 }
TRACe<x>:TXGain:FREeze?
```

Arguments OFF or 0 updates the display of the specified trace normally.

ON or 1 stops updating (freezes) the display of the specified trace.

Returns OFF or 0 means that the specified trace is updating normally on the display.

ON or 1 means that the specified trace is not updating on the display.

Examples `TRACE1:TXGAIN:COUNT:FREEZE ON` freezes Trace 1 in the display.

TRACe<x>:TXGain:FUNCtion

Sets or queries the trace processing method.

Conditions Measurement view: Transmission Gain

Where <x> is the trace and is 1, 2, or 3.

Group Display group

Syntax

```
TRACe<x>:TXGain:FUNCtion { NORMal | MAXHold | MINHold |
AVERage | AVGLog }
```

Arguments

NORMal selects the normal display.

AVERage selects the Average display that indicates the average signal level at each frequency point.

MAXHold selects the Max Hold display that indicates the maximum signal level at each frequency point.

MINHold selects the Min Hold display that indicates the minimum signal level at each frequency point.

AVGLog selects the Average of logs display that indicates the average signal level of logs at each frequency point.

Returns

See Arguments.

Examples

`TRACE1:TXGAIN:COUNT:FUNCTION MAXHold` selects Max Hold for Trace 1.

TRACe<x>:TXGain:SELect

Selects or queries the specified trace.

Conditions

Measurement view: Transmission Gain

Where <x> is the trace and is 1, 2, or 3.

Group

Display group

Syntax

```
TRACe<x>:TXGain:SELect
TRACe<x>:TXGain:SELect?
```

Arguments

None.

Returns

The currently selected trace number.

Examples

`TRACE1:TXGAIN:SELECT` selects Trace 2.

`TRACE1:TXGAIN:SELECT?` might return 1, indicating that Trace 1 is selected.

TRACe<x>:WLAN:CONSte

Turns the selected trace on or off in the constellation plot display, or queries the enabled status. < x> is accepted by the command but ignored as there is only a single WLAN constellation trace.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Trace commands

Syntax

```
TRACe<x>:WLAN:CONSte {0|1|OFF|ON}
TRACe<x>:WLAN:CONSte?
```

Arguments ON or 1 turns on the constellation trace.

OFF or 0 turns off the constellation trace.

Returns Returns the status of the selected trace as an integral value. 0 implies a disabled trace; 1 implies an enabled trace.

Examples TRAC:WLAN:CONS 1 turns the constellation trace on in the plot display.

TRACe<x>:WLAN:CONSte:FREeze

When the freeze state is turned on, the selected trace stops updating in the constellation plot. <x> is accepted by the command but ignored as there is only a single WLAN constellation trace.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Trace commands

Syntax

```
TRACe<x>:WLAN:CONSte:FREeze {0|1|OFF|ON}
TRACe<x>:WLAN:CONSte:FREeze?
```

Arguments ON or 1 turns on the freeze state, causing the constellation trace to stop updating.

OFF or 0 turns off the freeze state.

Returns Outputs the frozen status of the selected trace as an integral value. 1 indicates a frozen trace (stopped updating) and 0 a non-frozen trace.

Examples TRAC:WLAN:CONS:FRE 1 freezes the constellation trace, causing it to stop updating.

TRACe<x>:WLAN:CRESponse:SELect

Sets or queries which WLAN Channel Response trace is the active trace.TRACe1 is Magnitude, TRACe2 is Phase.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Trace commands

Syntax
```
TRACe<x>:WLAN:CRESponse:SELect
TRACe<x>:WLAN:CRESponse:SELect?
```

Arguments None

Returns Returns 1, which implies that the selected trace is the active trace. Returns 0 implies the selected trace is not the active trace.

Examples TRACE1:WLAN:CRES:SELECT specifies to select the magnitude trace.

TRACE1:WLAN:CRES:SELECT? might return 1, indicating that the magnitude trace is the active trace.

TRACe<x>:WLAN:EVM:FREeze

When the freeze state is turned on, the selected trace stops updating. TRACe1 is Averages. TRACe2 is Points.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Trace commands

Syntax
```
TRACe<x>:WLAN:EVM:FREeze {0|1|OFF|ON}
TRACe<x>:WLAN:EVM:FREeze?
```

Arguments ON or 1 turns on the freeze state, causing the selected trace to stop updating.

OFF or 0 turns off the freeze state.

Returns Outputs the frozen status of the selected trace as an integral value. 1 indicates a frozen trace and 0 a non-frozen trace.

Examples TRACE2:WLAN:EVM:FREEZE 1 causes Trace 2 to stop updating.

TRACE2:WLAN:EVM:FREEZE 1? might return 0, indicating that Trace 2 is not in a freeze state and is updating.

TRACe<x>:WLAN:EVM[:SHOW]

Displays or hides the selected trace. TRACe1 is Averages.TRACe2 is Points.

NOTE. *Use this command only with multicarrier standards and not singlecarrier (802.11b).*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Trace commands

Syntax
```
TRACe<x>:WLAN:EVM[:SHOW] {0|1|OFF|ON}
TRACe<x>:WLAN:EVM[:SHOW]?
```

Arguments ON or 1 displays the trace.

OFF or 0 hides the trace.

Examples `TRAC2:WLAN:EVM:SHOW 1` displays the WLAN EVM points trace.

TRACe<x>:WLAN:MERRor:FREeze

When the freeze state is turned on, the selected trace stops updating. `TRACe1` is Averages. `TRACe2` is Points.

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Trace commands

Syntax
```
TRACe<x>:WLAN:MERRor:FREeze {0|1|OFF|ON}
TRACe<x>:WLAN:MERRor:FREeze?
```

Arguments `ON` or `1` turns on the freeze state, causing the selected trace to stop updating.

`OFF` or `0` turns off the freeze state.

Returns Outputs the frozen status of the selected trace as an integral value. 1 indicates a frozen trace and 0 a non-frozen trace.

Examples `TRACE2:WLAN:MERR:FREEZE 1` causes Trace 2 to stop updating.

`TRACE2:WLAN:MERR:FREEZE 1?` might return 0, indicating that Trace 2 is not in a freeze state and is updating.

TRACe<x>:WLAN:MERRor[:SHOW]

Displays or hides the selected trace. `TRACe1` is Averages. `TRACe2` is Points.

NOTE. *Use this command only with multicarrier standards and not singlecarrier (802.11b).*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Trace commands

Syntax

```
TRACe<x>:WLAN:MERRor[:SHOW] {0|1|OFF|ON}
TRACe<x>:WLAN:MERRor[:SHOW]?
```

Arguments

ON or 1 displays the trace.

OFF or 0 hides the trace.

Examples

TRAC2:WLAN:MERR:SHOW 1 displays the WLAN Magnitude Error points trace.

TRACe<x>:WLAN:PERRor:FREeze

When the freeze state is turned on, the selected trace stops updating. TRACe1 is Averages. TRACe2 is Points.

Conditions

Measurement view: WLAN

This command requires WLAN Measurements

Group

Trace commands

Syntax

```
TRACe<x>:WLAN:PERRor:FREeze {0|1|OFF|ON}
TRACe<x>:WLAN:PERRor:FREeze?
```

Arguments

ON or 1 turns on the freeze state, causing the selected trace to stop updating.

OFF or 0 turns off the freeze state.

Returns

Outputs the frozen status of the selected trace as an integral value. 1 indicates a frozen trace and 0 a non-frozen trace.

Examples

TRACE2:WLAN:PERR:FREEZE 1 causes Trace 2 to stop updating.

TRACE2:WLAN:PERR:FREEZE 1? might return 0, indicating that Trace 2 is not in a freeze state and is updating.

TRACe<x>:WLAN:PERRor[:SHOW]

Displays or hides the selected trace. TRACe1 is Averages. TRACe2 is Points.

NOTE. *Use this command only with multicarrier standards and not singlecarrier (802.11b).*

Conditions Measurement view: WLAN

This command requires WLAN Measurements

Group Trace commands

Syntax
```
TRACe<x>:WLAN:PERRor[:SHOW] {0|1|OFF|ON}
TRACe<x>:WLAN:PERRor[:SHOW]?
```

Arguments ON or 1 displays the trace.

OFF or 0 hides the trace.

Examples TRAC2:WLAN:PERR:SHOW 1 displays the WLAN Phase Error points trace.

*TRG (No Query Form)

Generates a trigger. It produces the same effect as the Force Trigger button on the Trigger control panel. This command is valid when the trigger mode is Triggered.

Conditions Measurement views: All

Group IEEE common commands

Syntax *TRG

Arguments None

Examples *TRG generates a trigger.

TRIGger:ADVanced:SWEep:MODE

Sets or enables the analyzer to wait for a trigger for each acquisition in a swept spectrum or other measurement requiring multiple acquisitions worth of span.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Trigger group

Syntax `TRIGger:ADVanced:SWEep:MODE { OFF | ON | 0 | 1 }`

Arguments ON or 1 enables the wait for a trigger for each acquisition.

OFF or 0 disables the wait for a trigger for each acquisition.

Returns

Examples `TRIGGER:ADVANCED:SWEEP:MODE 1` causes the analyzer to wait for a trigger for each acquisition.

TRIGger:DPX:SHOW:FRAMes

Determines when to show frames in the swept acquisition mode. Show frames continuously when ON or show only triggered frames when OFF.

Conditions Measurement views: All

Group Trigger commands

Syntax
```
TRIGger:DPX:SHOW:FRAMes { OFF | ON | 0 | 1 }
TRIGger:DPX:SHOW:FRAMes?
```

Arguments `OFF` or `0` shows only triggered frames.

`ON` or `1` shows frames continuously.

Examples `TRIGger:DPX:SHOW:FRAMes ON` shows frames continuously.

TRIGger:DPX:TDM:SHOW:ALL

Sets or queries the selections of the DPX Prefs control panel tab page.

Conditions Measurement views: All

Group Trigger commands

Syntax

```
TRIGger:DPX:TDM:SHOW:ALL { OFF | ON | 0 | 1 }
TRIGger:DPX:TDM:SHOW:ALL?
```

Arguments

OFF or 0 selects the Synch with acquisition triggers option.

ON or 1 selects the Trigger as fast as possible option.

Examples

TRIGGER:DPX:TDM:SHOW:ALL ON selects the Trigger as fast as possible option in the DPX Prefs control panel tab.

TRIGger:EVENt:EXTernal:SLOPe

Sets or queries the signal slope of the External input to determine a trigger event.

Conditions

Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group

Trigger group

Syntax

```
TRIGger:EVENt:EXTernal:SLOPe { RISE | FALL }
```

Arguments

RISE: Trigger event occurs on the rising edge.

FALL: Trigger event occurs on the falling edge.

Examples

TRIGGER:EVENT:EXTERNAL:SLOPE RISE sets the trigger event to occur on the rising edge of the External signal input.

TRIGger:EVENt:INPut:LEVel

Sets or queries the power level that the signal must pass through to cause a trigger event when the trigger type is set to Power.

Conditions

Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group

Trigger group

Syntax

```
TRIGger:EVENt:INPut:LEVel <numeric>
```

Arguments <numeric>::=<NR3>

Range: –170 dBm to 50 dBm

Examples `TRIGGER:EVENT:INPUT:LEVEL 10` sets the trigger threshold level to 10 dBm.

TRIGger:EVENt:INPut:SLOPe

Sets or queries the signal slope of the RF Input to determine a trigger event.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Trigger group

Syntax `TRIGger:EVENt:INPut:SLOPe { RISE | FALL }`

Arguments RISE: Trigger event occurs on the rising edge.

FALL: Trigger event occurs on the falling edge.

Examples `TRIGGER:EVENT:INPUT:SLOPE RISE` sets the trigger event to occur on the rising edge of the RF Input signal.

TRIGger:EVENt:INPut:TYPE

Sets or queries the type of trigger. (Power is the only available setting.)

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Trigger group

Syntax `TRIGger:EVENt:INPut:TYPE {POWer}`

Arguments POWer: Sets trigger type to Power.

Examples `TRIGGER:EVENT:INPUT:TYPE POWER` sets the trigger type to Power.

TRIGger:EVENt:SOURce

Sets or queries the source of the signal to be monitored for a trigger event.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Trigger group

Syntax

```
TRIGger:EVENt:SOURce { INPut | EXTernal }
TRIGger:EVENt:SOURce?
```

Arguments INPut: Sets the trigger source to the RF Input.

EXTernal: Sets the trigger source to use the External input.

Examples `TRIGGER:EVENT:SOURCE INPUT` sets the trigger source to the RF Input.

TRIGger:MASK:NEW (No Query Form)

Loads a new frequency mask.

Conditions Measurement views: All

Group Trigger commands

Syntax

```
TRIGger:MASK:NEW <freq(1)>,<ampl(1)>,<freq(2)>,<ampl(2)>,
...,<freq(n)>,<ampl(n)>     (n = 500 maximum)
```

Arguments `<freq(n)>,<ampl(n)>` are numbers that represent a frequency (Hz) and amplitude (dBm) pair that define a point of the mask. Up to 500 pairs can be specified with zero frequency being the center screen. The mask is visible in the spectrum view with the following trigger conditions

NOTE. *Do not include a unit suffix (such as Hz, dBm, or dB) with the arguments for this command.*

- Trigger mode: Triggered (TRIGger[:SEQuence]:TIME:QUALified:TIME<x> is set to ON or 1.)
- Trigger type: Frequency Mask (TRIGger[:SEQuence]:EVENt:INPut:TYPE is set to FMASK.)

Examples `TRIGGER:MASK:NEW-8E6,-80,0,-10,8E6,-80` loads the mask with the points A (-8 MHz, -80 dBm), B (0 Hz, -10 dBm), and C (8 MHz, -80 dBm), as shown in the following figure.

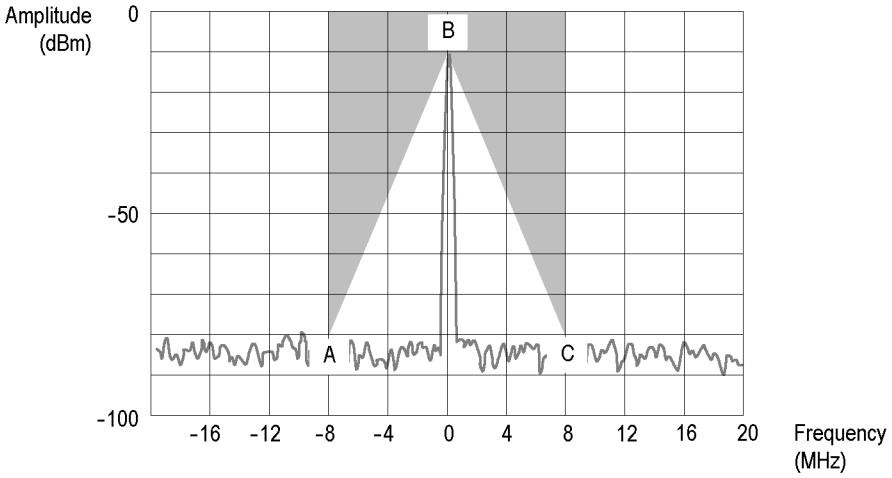


Figure 2-6: Trigger mask setting example

TRIGger:MASK:NEW:AUTO (No Query Form)

Draws a new frequency mask automatically based on a reference trace.

Conditions Measurement views: All

Group Trigger commands

Syntax
```
TRIGger:MASK:NEW:AUTO
<meas_ID>,<trace_ID>,<x_margin>,<y_margin>
```

Arguments `<meas_ID> :: <string>` specifies the measurement view.

`<trace_ID>` specifies the reference trace.

The values of `<meas_ID>` and `<trace_ID>` are listed in the following table for each possible measurement view.

<x_margin> :: <NRf> specifies the X margin (horizontal offset from the reference trace) in Hz.

<y_margin> :: <NRf> specifies the Y margin (vertical offset from the reference trace) in dB.

Measurement view	<meas_ID>	<trace_ID>
Spectrum	"specan"	TRACE1 (Trace 1), TRACE2 (Trace 2), TRACE3 (Trace 3), TRACE4 (Math trace), TRACE5 (Spectrogram trace)
DPX, Spectrum	"DPX"	TRACE1 (+Peak trace), TRACE2 (-Peak trace), TRACE3 (Avg trace), TRACE4 (Math trace)
Channel power and ACPR	"acpr"	TRACE1 (Trace 1)
MCPR	"mcpr"	TRACE1 (Trace 1)
OBW	"obw"	TRACE1 (Trace 1)

Examples TRIGGER:MASK:NEW:AUTO"specan",TRACE1,2E+6,15 draws a new frequency mask automatically in the Spectrum view based on Trace 1 with the horizontal margin of 2 MHz and the vertical margin of 15 dB.

TRIGger:MASK:OPEN (No Query Form)

Opens a trigger mask with a specified file. To save a trigger mask, use the TRIGger:MASK:SAVE command.

Conditions Measurement views: All

Group Trigger commands

Syntax TRIGger:MASK:OPEN <file_name>

Arguments <file_name> :: <string> specifies the trigger mask file to open. The file extension is .msk. You can omit the extension.

For the directory of file, refer to *Specifying the File* (See page 2-60.)

Examples TRIGGER:MASK:OPEN"C:\My Documents\Mask1" opens the trigger mask with the *Mask1* file in the *My Documents* directory.

TRIGger:MASK:SAVE (No Query Form)

Saves the current trigger mask to a specified file. To open the trigger mask, use the TRIGger:MASK:OPEN command.

Conditions Measurement views: All

Group Trigger commands

Syntax `TRIGger:MASK:SAVE <file_name>`

Arguments `<file_name> :: <string>` specifies the file to save the trigger mask. The file extension is .msk. You can omit the extension.

For the directory of file, refer to *Specifying the File* (See page 2-60.)

Examples `TRIGGER:MASK:SAVE"C:\My Documents\Mask1“` saves the trigger mask to the *Mask1* file in the *My Documents* directory.

TRIGger:SAVE:COUNt:MAXimum

Sets or queries the number of times a Save on Trigger action stores a file. After this limit is reached, no more files are saved. The instrument will continue to run, but no additional files are saved when a trigger event occurs.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Trigger group

Syntax `TRIGger:SAVE:COUNt:MAXimum <numeric>`

Arguments <numric>::=<NR1>

Range: 1 to 10000

Examples `TRIGGER:SAVE:COUNT:MAXIMUM 100` sets the number of files saved to 100.

TRIGger:SAVE:COUNt[:STATe]

Sets or queries the Maximum total number of files feature. When enabled and the set limit is reached, no more files are saved. The instrument will continue to run, but no additional files are saved when a trigger event occurs.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Trigger group

Syntax `TRIGger:SAVE:COUNt[:STATe] { OFF | ON | 0 | 1 }`

Arguments ON or 1 enables the maximum number of saved files feature.

OFF or 0 disables the maximum number of saved files feature.

Examples `TRIGGER:SAVE:COUNT[:STATE] 1` enables the maximum number of saved files feature.

TRIGger:SAVE:DATA:FORMat

Sets or queries the file format used when saving the acquisition data on a trigger event.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Trigger group

Syntax `TRIGger:SAVE:DATA:FORMat { TIQ | CSV | MAT }`

Arguments TIQ: Native format data file which may include multiple frames.

CSV: Comma-separated values format.

MAT: MATLAB format.

Examples `TRIGGER:SAVE:DATA:FORMAT TIQ` saves the data file in native format.

TRIGger:SAVE:DATA[:STATe]

Sets or queries to save data when a trigger event occurs.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Trigger group

Syntax `TRIGger:SAVE:DATA[:STATe] { OFF | ON | 0 | 1 }`

Arguments ON or 1 enables the save on trigger function.

OFF or 0 disables the save on trigger function

Examples `TRIGGER:SAVE:DATA[:STATE] 1` enables the save on trigger function.

TRIGger:SAVE:PICTure:FORMat

Sets or queries the file format used when saving a picture of the display when a trigger event occurs.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Trigger group

Syntax `TRIGger:SAVE:PICTure:FORMat { PNG | JPG | BMP }`

Arguments PNG, JPG, BMP: picture formats available.

Examples `TRIGGER:SAVE:PICTURE:FORMAT PNG` sets the picture format extension to .PNG.

TRIGger:SAVE:PICTure[:STATe]

Sets or queries the save picture on trigger setting. This enables saving a picture of the display when a trigger event occurs.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Trigger group

Syntax `TRIGger:SAVE:PICTure[:STATe] { OFF | ON | 0 | 1 }`

Arguments ON or 1 enables saving a picture.

OFF or 0 disables saving a picture.

Examples `TRIGGER:SAVE:PICTURE[:STATE] 1` enables saving a picture of the display when a trigger event occurs.

TRIGger[:SEQuence]:ADVanced:HOLDoff

Sets or queries the trigger holdoff time which prevents triggers until there have been no trigger events for at least the specified holdoff period.

Conditions Measurement views: All

Group Trigger commands

Syntax
```
TRIGger[:SEQuence]:ADVanced:HOLDoff <value>
TRIGger[:SEQuence]:ADVanced:HOLDoff?
```

Arguments `<value> ::= <NRf>` specifies the trigger holdoff time. Range: 20 ns to 10 s.

Examples `TRIGger[:SEQuence]:ADVanced:HOLDoff 1.5us` sets the trigger holdoff time to 1.5 µs.

TRIGger[:SEQuence]:ADVanced:HOLDoff:ENABle

Enables or queries the status of the Advanced Trigger Holdoff function.

Conditions Measurement views: All

Group Trigger commands

Syntax

```
TRIGger[:SEQuence]:ADVanced:HOLDoff:ENABle { OFF | ON | 0 | 1 }
TRIGger[:SEQuence]:ADVanced:HOLDoff:ENABle?
```

Arguments

`OFF` or `0` does not trigger based on the advanced trigger holdoff settings.

`ON` or `1` enables triggering based on the advanced trigger holdoff settings.

Examples

`TRIGger:ADVanced:HOLDoff:ENABle ON` enables triggering based on the advanced trigger holdoff settings.

TRIGger[:SEQuence]:ADVanced:SWEep:MODE

Determines whether or not to trigger each segment in the swept acquisition mode.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:ADVanced:SWEep:MODE { OFF | ON | 0 | 1 }
TRIGger[:SEQuence]:ADVanced:SWEep:MODE?
```

Arguments

`OFF` or `0` does not trigger each segment in the swept acquisition mode.

`ON` or `1` triggers each segment in the swept acquisition mode.

Examples

`TRIGGER:SEQUENCE:ADVANCED:SWEEP:MODE ON` triggers each segment in the swept acquisition mode.

TRIGger[:SEQuence]:ADVanced:SWEPt:SEGMent:ENABle

Determines whether or not to set the spectrum analyzer to wait for a trigger for each acquisition in a swept spectrum or other measurement requiring multiple acquisitions worth of span.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:ADVanced:SWEPt:SEGMent:ENABle { OFF | ON
| 0 | 1 }
TRIGger[:SEQuence]:ADVanced:SWEPt:SEGMent:ENABle?
```

Arguments

`OFF` or `0` does not trigger each segment in the swept acquisition mode.

`ON` or `1` triggers each segment in the swept acquisition mode.

Examples

`TRIGger[:SEQuence]:ADVanced:SWEPt:SEGMent:ENABle ON` waits for a trigger for each acquisition in a swept spectrum or other measurement requiring multiple acquisitions worth of span.

TRIGger[:SEQuence]:EVENt:EXTFront:IMPedance

Sets or queries the impedance of the external trigger input on the front panel.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:EXTFront:IMPedance { OHM50 | OHM5K
}
TRIGger[:SEQuence]:EVENt:EXTFront:IMPedance?
```

Arguments

`OHM50` selects 50Ω impedance.

`OHM5K` selects 5 kΩ impedance.

Examples

`TRIGGER:SEQUENCE:EVENT:EXTFRONT:IMPEDANCE OHM50` selects 50Ω impedance for the external trigger input on the front panel.

TRIGger[:SEQuence]:EVENt:EXTFront:LEVel

Sets or queries the trigger level at the external trigger input on the front panel.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:EXTFront:LEVel <value>
TRIGger[:SEQuence]:EVENt:EXTFront:LEVel?
```

Arguments

<value> :: <NRf> specifies the trigger level. Range: -2.5 to +2.5 V.

Examples

`TRIGGER:SEQUENCE:EVENT:EXTFRONT:LEVEL 1.5` sets the trigger level to 1.5 V at the external trigger input on the front panel.

TRIGger[:SEQuence]:EVENt:EXTFront:SLOPe

Sets or queries the trigger slope of the external trigger input on the front panel.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:EXTFront:SLOPe { RISe | FALL }
TRIGger[:SEQuence]:EVENt:EXTFront:SLOPe?
```

Arguments

`RISe` causes the trigger event on the rising edge.

`FALL` causes the trigger event on the falling edge.

Examples

`TRIGGER:SEQUENCE:EVENT:EXTFRONT:SLOPE RISe` causes the trigger event on the rising edge of the external input signal.

TRIGger[:SEQuence]:EVENt:EXTRear:SLOPe

Sets or queries the trigger slope of the external trigger input on the rear panel. The trigger level is fixed to the TTL threshold.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:EXTRear:SLOPe { RISe | FALL }
TRIGger[:SEQuence]:EVENt:EXTRear:SLOPe?
```

Arguments `RISe` causes the trigger event on the rising edge.

`FALL` causes the trigger event on the falling edge.

Examples `TRIGGER:SEQUENCE:EVENT:EXTREAR:SLOPE RISe` causes the trigger event on the rising edge of the external input signal.

TRIGger[:SEQuence]:EVENt:GATed

Sets or queries the positive or negative logic for the gated trigger input on the rear panel.

Conditions Measurement views: All

Group Trigger commands

Syntax
```
TRIGger[:SEQuence]:EVENt:GATed { HIGh | LOW }
TRIGger[:SEQuence]:EVENt:GATed?
```

Arguments `HIGh` specifies that the gated trigger input is high active.

`LOW` specifies that the gated trigger input is low active.

Examples `TRIGGER:SEQUENCE:EVENT:GATED HIGh` specifies that the gated trigger input is high active.

TRIGger[:SEQuence]:EVENt:INPut:DDENsity:AMPLitude

Sets or queries the center vertical level for the density trigger region of the RF input DPX bitmap display.

Conditions Measurement views: DPX bitmap

Group Trigger commands

Syntax
```
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:AMPLitude <value>
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:AMPLitude?
```

Related Commands TRIGger[:SEQuence]:EVENt:INPut:DDENsity:AMPLitude:TOLerance

Arguments `<value> :: <NRf>` specifies the vertical center of the trigger region in dBm. Range: 0 to 200 dBm.

Examples `TRIGger[:SEQuence]:EVENt:INPut:DDENsity:AMPLitude 100dBm` sets the center level to 100 dBm for the density trigger region.

TRIGger[:SEQuence]:EVENt:INPut:DDENsity:AMPLitude:TOLerance

Sets or queries the positive and negative extensions of the density trigger region from the center vertical level. The density trigger region is defined for the DPX bitmap display.

Conditions Measurement views: DPX bitmap

Group Trigger commands

Syntax
```
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:AMPLitude:TOLerance
<value>
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:AMPLitude:TOLerance?
```

Related Commands TRIGger[:SEQuence]:EVENt:INPut:DDENsity:FREQuency:TOLerance

Arguments `<value> :: <NRf>` specifies an absolute amplitude value for the positive and negative limits of the density trigger region in dBm. Range: within the defined measurement box.

Examples `TRIGger[:SEQuence]:EVENt:INPut:DDENsity:AMPLitude:TOLerance 50dBm` sets the positive limit 50 dBm above the AMPLItude level and the negative limit 50 dBm below the AMPLItude level.

TRIGger[:SEQuence]:EVENt:INPut:DDENsity:FREQuency

Sets or queries the center horizontal level for the density trigger region of the RF input DPX bitmap display.

Conditions Measurement views: DPX bitmap

Group Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:FREQuency <value>
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:FREQuency?
```

Related Commands

Arguments <value> :: <NRf> specifies the horizontal center of the trigger region. Range: within the defined measurement box.

Examples TRIGger[:SEQuence]:EVENt:INPut:DDENsity:FREQuency 1.82GHz sets the center horizontal level to 1.82 GHz for the density trigger region.

TRIGger[:SEQuence]:EVENt:INPut:DDENsity:FREQuency:TOLerance

Sets or queries the left and right extensions of the density trigger region from the center horizontal level. The density trigger region is defined for the DPX bitmap display.

Conditions Measurement views: DPX bitmap

Group Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:FREQuency:TOLerance
<value>
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:FREQuency:TOLerance?
```

Related Commands

Arguments <value> :: <NRf> specifies an absolute frequency value for the left and right limits from the center frequency of the density trigger region. Range: within the defined measurement box.

Examples TRIGger[:SEQuence]:EVENt:INPut:DDENsity:FREQuency:TOLerance 0.2GHz, for a center frequency of 1.82 GHz, sets the left horizontal limit to 1.62 GHz and the right horizontal limit to 2.02 GHz for the density trigger region.

TRIGger[:SEQuence]:EVENt:INPut:DDENsity:THReshold

Sets or queries the DPX Density threshold that defines a trigger event for the density trigger region of the DPX bitmap display. The DPX Density must be higher or lower (specified by the Density selection) than this value to define a trigger event.

Conditions Measurement views: DPX bitmap

Group Trigger commands

Syntax
```
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:THReshold <value>
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:THReshold?
```

Related Commands

Arguments `<value> ::  <NRf>` specifies the DPX Density threshold as a percentage (%). Range: within the defined measurement box.

Examples `TRIGger[:SEQuence]:EVENt:INPut:DDENsity:THReshold 2` sets the DPX Density threshold to 2 % for the density trigger.

TRIGger[:SEQuence]:EVENt:INPut:DDENsity:VIOLation

Sets or queries whether a density value higher or lower than the THReshold value defines a trigger event for the density trigger region of the DPX bitmap display.

Conditions Measurement views: DPX bitmap

Group Trigger commands

Syntax
```
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:VIOLation { HIGHer
| LOWer }
TRIGger[:SEQuence]:EVENt:INPut:DDENsity:VIOLation?
```

Related Commands TRIGger[:SEQuence]:EVENt:INPut:DDENsity:THReshold

Arguments `HIGHer` a DPX density value above the THReshold value defines the trigger event.

`LOWer` a DPX Density value below the THReshold value defines the trigger event.

Examples `TRIGger:EVENt:INPut:DDENsity:VIOLation LOWER` sets the DPX Density VIOLATION to trigger when the probability of occurrence is lower than the THRESHOLD percentage.

TRIGger[:SEQuence]:EVENt:INPut:FEDGe:LEVel

Sets or queries the trigger level at the frequency edge input.

Conditions Measurement views: All

Group Trigger commands

Syntax
```
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:LEVel <value>
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:LEVel?
```

Arguments <value> :: <NRf> specifies the trigger level.

Range: ± (Max AcqBW/2) Hz

Examples `TRIGGER:SEQUENCE:EVENT:INPUT:FEDGE:LEVEL 1.5` sets the trigger level to 1.5 V at the frequency edge input.

TRIGger[:SEQuence]:EVENt:INPut:FEDGe:LEVel:ACTual? (Query Only)

Queries the actual length of the frequency edge input for the DPX measurement.

Conditions Measurement views: All

Group Trigger commands

Syntax
```
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:LEVel:ACTual?
```

Arguments None

Returns <NRf> The actual length in Hz.

Examples `TRIGGER:SEQUENCE:EVENT:INPUT:FEDGE:LEVEL:ACTUAL?` might return 10 Hz, indicating that the actual length is 10 Hz.

TRIGger[:SEQuence]:EVENt:INPut:FEDGe:SLOPe

Sets or queries the trigger slope of the frequency edge input.

Conditions Measurement views: All

Group Trigger commands

Syntax
```
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:SLOPe { RISe | FALL }
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:SLOPe?
```

Arguments `RISe` causes the trigger event on the rising edge.

`FALL` causes the trigger event on the falling edge.

Examples `TRIGGER:SEQUENCE:EVENT:INPUT:FEDGE:SLOPE RISE` causes the trigger event on the rising edge of the frequency edge input.

TRIGger[:SEQuence]:EVENt:INPut:FEDGe:THReshold:LEVel

Set or queries the threshold level in the DPX measurement.

Conditions Measurement views: All

Group Trigger commands

Syntax
```
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:THReshold:LEVel <value>
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:THReshold:LEVel?
```

Arguments `<value> ::  <NRf>` specifies the threshold level.

Range: 50 dBm to –170 dBm

Examples `TRIGGER:SEQUENCE:EVENT:INPUT:FEDGE:THRESHOLD:LEVEL –25` sets the threshold level to –25 dBm.

TRIGger[:SEQuence]:EVENt:INPut:FEDGe:THReshold:STATe

Set or queries the threshold state in the DPX measurement.

Conditions Measurement views: All

Group Trigger commands

Syntax
```
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:THReshold:STATe { OFF | ON | 0 | 1 }
TRIGger[:SEQuence]:EVENt:INPut:FEDGe:THReshold:STATe?
```

Arguments `OFF` or `0` disables the threshold.

`ON` or `1` enables the threshold.

Examples `TRIGGER:SEQUENCE:EVENT:INPUT:TDBWIDTH:STATE OFF` enables the threshold.

TRIGger[:SEQuence]:EVENt:INPut:FMASk:{BANDwidth|BWIDth}[:RESolution]

Sets or queries the Resolution Bandwidth value to be used in the spectrum view for the frequency mask trigger.

Conditions Measurement views: All

Group Trigger commands

Syntax
```
TRIGger[:SEQuence]:EVENt:INPut:FMASk:{BANDwidth|BWIDth}[:RESolution] <value>
TRIGger[:SEQuence]:EVENt:INPut:FMASk:{BANDwidth|BWIDth}[:RESolution]?
```

Arguments `<value> ::= <NRf>` specifies the resolution bandwidth for Frequency Mask triggering. Range: .

Examples `TRIGger[:SEQuence]:EVENt:INPut:FMASk:BANDwidth 300kHz` sets the resolution bandwidth to 300 kHz.

TRIGger[:SEQuence]:EVENt:INPut:FMASk:BANDwidth|BWIDth[:RESolution]:ACTual? (Query Only)

Queries current setting for the Resolution Bandwidth value used in the spectrum view for the frequency mask trigger.

Conditions Measurement views: All

Group Trigger commands

Syntax
```
TRIGger[:SEQuence]:EVENt:INPut:FMASk:BANDwidth|BWIDth[:
RESolution]:ACTual?
```

Arguments None.

Examples `TRIGger:EVENt:INPut:FMASk:BWIDth:ACTUAL?` returns the resolution bandwidth value.

TRIGger[:SEQuence]:EVENt:INPut:FMASk:BANDwidth|BWIDth[:RESolution]:AUTO

Sets or queries whether to automatically set the Resolution Bandwidth value used in the spectrum view for the frequency mask trigger.

Conditions Measurement views: All

Group Trigger commands

Syntax
```
TRIGger[:SEQuence]:EVENt:INPut:FMASk:BANDwidth|BWIDth[:
RESolution]:AUTO { OFF | ON | 0 | 1 }
TRIGger[:SEQuence]:EVENt:INPut:FMASk:BANDwidth|BWIDth[:
RESolution]:AUTO?
```

Arguments `OFF` or `0` sets the resolution bandwidth manually with the command TRIGger[:SEQuence]:EVENt:INPut:FMASk:{BANDwidth|BWIDth}[:RESolution].

`ON` or `1` sets the resolution bandwidth automatically.

Examples `TRIGger:EVENt:INPut:FMASk:BWIDth:AUTO ON` sets the resolution bandwidth automatically.

TRIGger[:SEQuence]:EVENt:INPut:FMASk:VIOLation

Sets or queries when the analyzer triggers in the frequency mask trigger.

Conditions Measurement views: All

Group Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:FMASk:VIOLation { T | F | TF | FT | TFT | FTF }
TRIGger[:SEQuence]:EVENt:INPut:FMASk:VIOLation?
```

Arguments The following table lists the arguments which represent the trigger violations.

Table 2-38: Trigger violations

Violation	Description
T	Only one state is required to initiate a trigger event. The signal has at least one data point inside the mask. The trigger event occurs at the first point that appears inside the mask. A trigger event could occur on the first acquisition.
F	Only one state change is required to initiate a trigger event. The signal has at least one data point outside the mask. The trigger event occurs at the first point that appears outside the mask. A trigger event could occur on the first acquisition.
TF	Two states are required to initiate a trigger event. The signal must be inside the mask and then passes out of the mask. The trigger event occurs at the first transition where the signal passes out of the mask.
FT	Two states are required to initiate a trigger event. The signal must be outside the mask and then passes into the mask. The trigger event occurs at the first transition where the signal passes into the mask.
TFT	Three states are required to initiate a trigger event. The signal starts inside the mask and then passes out of the mask. Next, the signal must pass into the mask. The trigger event occurs at the second transition where the signal passes back into the mask.
FTF	Three states are required to initiate a trigger event. The signal starts outside the mask and then passes into the mask. Next, the signal must pass back outside the mask. The trigger event occurs at the second transition where the signal passes back out of the mask.

Examples `TRIGGER:SEQUENCE:EVENT:INPUT:FMASK:VIOLATION TF` specifies that the analyzer will trigger when the signal has crossed into the mask and then outside of the mask.

TRIGger[:SEQuence]:EVENt:INPut:LEVel

Sets or queries the trigger level for the RF input level trigger.

Conditions Measurement views: All

Group Trigger commands

Syntax
```
TRIGger[:SEQuence]:EVENt:INPut:LEVel <value>
TRIGger[:SEQuence]:EVENt:INPut:LEVel?
```

Arguments <value> :: <NRf> specifies the trigger level. Range: -170 to +50 dBm.

Examples TRIGGER:SEQUENCE:EVENT:INPUT:LEVEL -10 sets the trigger level to -10 dBm for the RF input level trigger.

TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULSe

Sets or queries the Runt trigger for a positive or negative going pulse.

Conditions Measurement views: All

Group Trigger commands

Syntax
```
TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULSe { POSitive |
NEGative }
TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULSe?
```

Arguments POSitive causes the trigger event on a positive runt pulse.

NEGative causes the trigger event on a negative runt pulse.

Examples TRIGger:EVENt:INPut:RUNT:PULSE POSitive causes the trigger event on a positive pulse.

TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULse:HIGH:LEVel

Sets or queries the upper level (first threshold) to qualify a Runt trigger.

Conditions Measurement views: All

Group Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULSe:HIGH:LEVel <value>
TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULSe:HIGH:LEVel?
```

Arguments

<value> :: <NRf> sets the upper vertical level for a runt pulse in dBm.

Examples

TRIGger:EVENt:INPut:RUNT:PULSE 300 causes the trigger event when the leading edge of a pulse crosses the 300 dBm level.

TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULSe:LOW:LEVel

Sets or queries the lower level (second threshold) to qualify a Runt trigger.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULSe:LOW:LEVel <value>
TRIGger[:SEQuence]:EVENt:INPut:RUNT:PULSe:LOW:LEVel?
```

Arguments

<value> :: <NRf> sets the lower vertical level for a runt pulse in dBm.

Examples

TRIGger:EVENt:INPut:RUNT:PULSE 3 causes the trigger event when the following edge of a pulse crosses the 3 dBm level.

TRIGger[:SEQuence]:EVENt:INPut:SLOPe

Sets or queries the trigger slope for the RF input level trigger.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:SLOPe { RISe | FALL }
TRIGger[:SEQuence]:EVENt:INPut:SLOPe?
```

Arguments

RISe causes the trigger event on the rising edge.

FALL causes the trigger event on the falling edge.

Examples `TRIGGER:SEQUENCE:EVENT:INPUT:SLOPE RISe` causes the trigger event on the rising edge of the RF input signal.

TRIGger[:SEQuence]:EVENt:INPut:TDBWidth

Sets or queries the time-domain bandwidth for the RF input power trigger. This command is effective when TRIGger[:SEQuence]:EVENt:INPut:TDBWidth:STATe is ON.

Conditions Measurement views: All

Group Trigger commands

Syntax
```
TRIGger[:SEQuence]:EVENt:INPut:TDBWidth <value>
TRIGger[:SEQuence]:EVENt:INPut:TDBWidth?
```

Arguments `<value> :: <NRf>` specifies the time-domain bandwidth.
Range: 1 Hz to 60 MHz.

Examples `TRIGGER:SEQUENCE:EVENT:INPUT:TDBWIDTH 5MHz` sets the time-domain bandwidth to 5 MHz for the RF input power trigger.

TRIGger[:SEQuence]:EVENt:INPut:TDBWidth:ACTual? (Query Only)

Queries the actual time-domain bandwidth for the RF input power trigger.

Conditions Measurement views: All

Group Trigger commands

Syntax
```
TRIGger[:SEQuence]:EVENt:INPut:TDBWidth:ACTual?
```

Related Commands TRIGger[:SEQuence]:EVENt:INPut:TDBWidth

Arguments None

Returns `<NRf>` Actual time-domain bandwidth.

Examples `TRIGGER:SEQUENCE:EVENT:INPUT:TDBWIDTH:ACTUAL?` might return `1.000E+6`, indicating that the actual time-domain bandwidth is 1 MHz.

TRIGger[:SEQuence]:EVENt:INPut:TDBWidth:STATe

Determines whether to set the time-domain bandwidth automatically or manually for the RF input power trigger.

Conditions Measurement views: All

Group Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:TDBWidth:STATe { OFF | ON |
0 | 1 }
TRIGger[:SEQuence]:EVENt:INPut:TDBWidth:STATe?
```

Arguments `OFF` or `0` sets the time-domain bandwidth automatically.

`ON` or `1` sets the time-domain bandwidth manually using the TRIGger[:SEQuence]:EVENt:INPut:TDBWidth command.

Examples `TRIGGER:SEQUENCE:EVENT:INPUT:TDBWIDTH:STATE OFF` sets the time-domain bandwidth automatically.

TRIGger[:SEQuence]:EVENt:INPut:TYPE

Sets or queries the trigger type for the source of RF input.

Conditions Measurement views: All

Group Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:INPut:TYPE { POWer | FMASK |
DDENsity | RUNT | FEDGe }
TRIGger[:SEQuence]:EVENt:INPut:TYPE?
```

Arguments `POWer` uses the power level for triggering.

`FMASK` uses the frequency mask for triggering.

DDENsity uses the density mask for triggering.

RUNT uses the runt mask for triggering.

Examples TRIGGER:SEQUENCE:EVENT:INPUT:TYPE FMASK uses the frequency mask for triggering.

TRIGger[:SEQuence]:EVENt:SOURce

Sets or queries the trigger event source.

Conditions Measurement views: All

Group Trigger commands

Syntax

```
TRIGger[:SEQuence]:EVENt:SOURce { INPut | EXTFront | EXTRear
| EXTGated | LINe }
TRIGger[:SEQuence]:EVENt:SOURce?
```

Arguments The following table lists the arguments.

Table 2-39: Trigger event source

Argument	Source
INPut	RF input
EXTFront	Trigger in (front)
EXTRear	Trigger in (rear)
EXTGated	Gate in
LINe	AC line

Examples TRIGGER:SEQUENCE:EVENT:SOURCE INPut specifies the trigger event source as the RF input.

TRIGger[:SEQuence]:FORCed

Determines whether or not to cause a manual trigger if the acquisition is armed, ready and waiting for a trigger. This command is valid when TRIGger[:SEQuence]:TIME:QUALified:TIME<x> is set to On (the trigger mode is Triggered).

Conditions Measurement views: All

Group Trigger commands

Syntax
```
TRIGger[:SEQuence]:FORCed { OFF | ON | 0 | 1 }
TRIGger[:SEQuence]:FORCed?
```

Arguments `OFF` or `0` does not cause a manual trigger.

`ON` or `1` causes a manual trigger.

Examples `TRIGGER:SEQUENCE:FORCED ON` causes a manual trigger if the acquisition is armed, ready and waiting for a trigger.

TRIGger:SEQuence:FORCed (No Query Form)

Causes a trigger event to occur when the trigger mode is set to Triggered, regardless of the set trigger conditions.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Trigger group

Syntax `TRIGger:SEQuence:FORCed`

Arguments None

Examples `TRIGGER:SEQUENCE:FORCED` forces a trigger event.

TRIGger[:SEQuence]:IMMediate (No Query Form)

Causes a trigger immediately, skipping the event detection and delay. This command is valid when TRIGger[:SEQuence]:TIME:QUALified:TIME<x> is set to On (the trigger mode is Triggered).

Conditions Measurement views: All

Group Trigger commands

Syntax
```
TRIGger[:SEQuence]:IMMediate
```

Arguments None

Examples `TRIGGER:SEQUENCE:IMMEDIATE` causes a trigger immediately, skipping the event detection and delay.

TRIGger[:SEQuence]:STATus

Sets or queries the trigger mode (Free Run or Triggered).

Conditions Measurement views: All

Group Trigger commands

Syntax
```
TRIGger[:SEQuence]:STATus { OFF | ON | 0 | 1 }
TRIGger[:SEQuence]:STATus?
```

Arguments `OFF` or `0` selects the free-run mode.

`ON` or `1` selects the triggered mode.

Examples `TRIGGER:SEQUENCE:STATUS ON` selects the triggered mode.

TRIGger[:SEQuence][:STATus]

Sets or queries the trigger mode setting.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Trigger group

Syntax
```
TRIGger[:SEQuence][:STATus] { 0 | 1 }
TRIGger[:SEQuence][:STATus]?
```

Arguments

0: sets the trigger mode to Free Run.

1: sets the trigger mode to Triggered.

Examples

`TRIGGER[:SEQUENCE][:STATUS] 0` sets the trigger mode to Free Run.

TRIGger[:SEQuence]:TIME:DELay

Sets or queries the trigger delay time which occurs after recognizing an event but before actually declaring the trigger.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:TIME:DELay <value>
TRIGger[:SEQuence]:TIME:DELay?
```

Arguments

`<value> ::  <NRf>` specifies the trigger delay time. Range: 0 to 60 s.

Examples

`TRIGGER:SEQUENCE:TIME:DELAY 1.5` sets the trigger delay time to 1.5 s.

TRIGger[:SEQuence]:TIME:POSition

Sets or queries the position of the trigger event within the acquisition record.

Conditions

Measurement views: All

Group

Trigger commands

Syntax

```
TRIGger[:SEQuence]:TIME:POSition <value>
TRIGger[:SEQuence]:TIME:POSition?
```

Arguments

`<value> ::  <NRf>` specifies the trigger position. Range: 1% to 99%.

Examples

`TRIGGER:SEQUENCE:TIME:POSITION 20` sets the trigger position to 20% of the acquisition record.

TRIGger[:Sequence]:TIME:POSition:AUTO

Sets or queries the position of the trigger event (automatic or manual) within the acquisition record.

Conditions Measurement views: All

Group Trigger commands

Syntax

```
TRIGger[:Sequence]:TIME:POSition:AUTO { OFF | ON | 0 | 1 }
TRIGger[:Sequence]:TIME:POSition:AUTO?
```

Arguments `OFF` or `0` sets the position of the trigger event manually.

`ON` or `1` sets the position of the trigger event automatically.

Examples `TRIGGER:SEQUENCE:TIME:POSITION:AUTO ON` sets the position of the trigger event automatically.

TRIGger[:SEQuence]:TIME:QUALified

Sets or queries the timing qualification setting for triggers. Trigger events that do not meet the timing qualifications are ignored.

Conditions Measurement views: All

Group Trigger commands

Syntax

```
TRIGger[:SEQuence]:TIME:QUALified { SHORter | LONGer |
INSide | OUTSide | NONE }
TRIGger[:SEQuence]:TIME:QUALified?
```

Arguments `SHORter` the length of the trigger event is shorter than the time specified by the TRIGger:TIME:QUALified:TIME1 setting.

`LONGer` the length of the trigger event is longer than the time specified by the TRIGger:TIME:QUALified:TIME1 setting.

`INSide` the length of the trigger event is longer than the time TRIGger:TIME:QUALified:TIME1 setting and shorter than the TRIGger:TIME:QUALified:TIME2 setting.

`OUTSide` the length of the trigger event is shorter than the time TRIGger:TIME:QUALified:TIME1 setting or longer than the TRIGger:QUALified:TIME2 setting.

Examples `TRIGger:SEQuence:TIME:QUALified SHORter` the length of the trigger event is shorter than the time specified by the TRIGger:TIME:DELAY setting

TRIGger[:SEQuence]:TIME:QUALified:TIME<x>

Sets or queries the Time1 and Time2 parameters for time qualified triggering. Time1 is the shorter or first time, and Time2 is the second or later time for the qualifying period. Each TIME<x> must be set with a separate command.

Conditions Measurement views: All

Group Trigger commands

Syntax `TRIGger[:SEQuence]:TIME:QUALified:TIME<x> <value>`

Related Commands TRIGger[:SEQuence]:TIME:QUALified

Arguments `<value> :: <NRf>` the length of the trigger time for TIME1 or TIME2. Range: 0 s to 28 s.

Examples `TRIGger:TIME:QUALified:TIME1 55us` sets the TIME1 trigger point at 55 us after the trigger event.

TRIGger:TIME:POSition

Sets or queries the point (in percent) along the acquisition record the trigger event should occur. Manually setting the trigger position disables the Auto setting.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Trigger group

Syntax `TRIGger:TIME:POSition <numeric>`

Arguments <numeric>::=<NR1>

Range: 1 to 100

Examples `TRIGGER:TIME:POSITION 50` sets the acquisition point to 50%.

TRIGger:TIME:POSition:AUTO

Sets or queries the instrument to automatically set the position of the trigger event.

Conditions Requires an RSA 306, 306B, 500A series, or 600A series be connected.

Group Trigger group

Syntax

```
TRIGger:TIME:POSition:AUTO { OFF | ON | 0 | 1 }
```

Arguments ON or 1 enables auto trigger position.

OFF or 0 disables auto trigger position.

Examples `TRIGGER:TIME:POSITION:AUTO 1` enables the auto trigger position setting.

UNIT:POWer

Sets or queries the fundamental unit of power. This command is equivalent to [SENSe]:POWer:UNITs.

Conditions Measurement views: All

Group Unit commands

Syntax

```
UNIT:POWer { DBM | DBV | VOLTs | WATTs | DBUW | DBW| DBUV |
DBMV | DBUA | DBUV_M | DBUA_M | AMPS }
UNIT:POWer?
```

Arguments The following table lists the arguments.

Table 2-40: Power units

Argument	Power unit
DBM	dBm
DBV	dBV
VOLTs	Volts
WATTs	Watts
DBUW	dBµW
DBW	dBW
DBUV	dBµV
DBMV	dBmV
DBUA	dBµA
DBUV_M	dBµV/m
DBUA_M	dBµA/m
AMPS	Amps

NOTE. *Select dBµV/m or dBµA/m unit when using an antenna table.*

Examples `UNIT:POWERDBM` specifies the fundamental unit of power as dBm.

*WAI (No Query Form)

Prevents the analyzer from executing further commands or queries until all pending operations finish. This command allows you to synchronize the operation of the analyzer with your application program. For the details, refer to *Synchronizing Execution* (See page 3-11.).

Conditions Measurement views: All

Group IEEE common commands

Syntax `*WAI`

Related Commands *OPC

Arguments None

Status and Events

Status and Events

The SCPI interface in the analyzer includes a status and event reporting system that enables the user to monitor crucial events that occur in the instrument. The analyzer is equipped with four registers and one queue that conform to IEEE Std 488.2-1987. This section will discuss these registers and queues along with status and event processing.

Status and Event Reporting System

The following figure outlines the status and event reporting mechanism offered in the RSA Series analyzers. It contains three major blocks

- Standard Event Status
- Operation Status
- Questionable Status (fan-out structure)

The processes performed in these blocks are summarized in the Status Byte. The three blocks contain four types of registers as shown in the following table.

Table 3-1: Register type

Register	Description
Condition register	Records event occurrence in the instrument. Read only.
Transition register (positive/negative)	A positive transition filter allows an event to be reported when a condition changes from false to true. A negative filter allows an event to be reported when a condition changes from true to false. Setting both positive and negative filters true allows an event to be reported anytime the condition changes. Clearing both filters disables event reporting.
Event register	Records events filtered by the transition register. Read only.
Enable register	Masks the event register to report in the summary bit. User-definable.

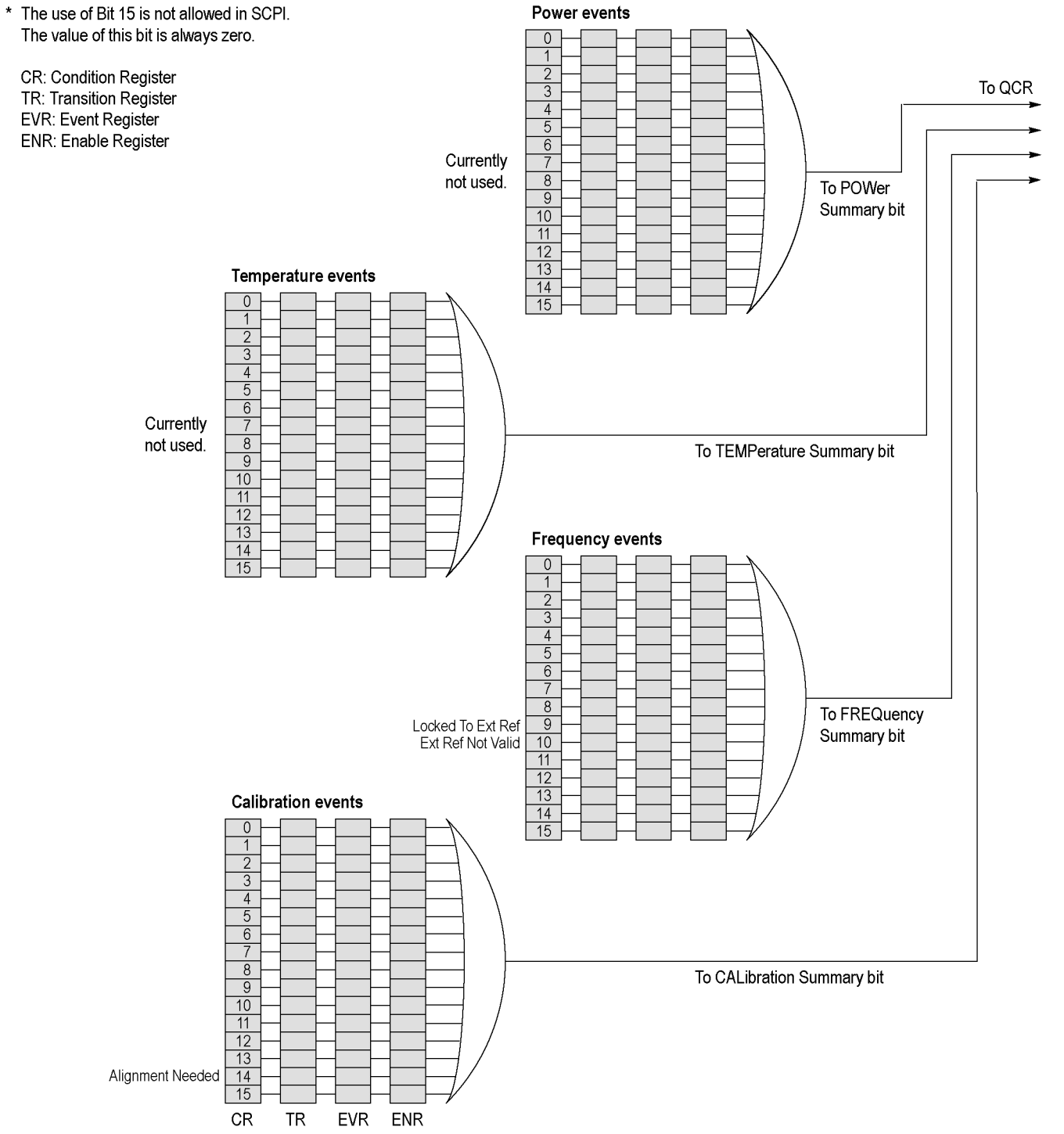


Figure 3-1: Status/Event reporting mechanism

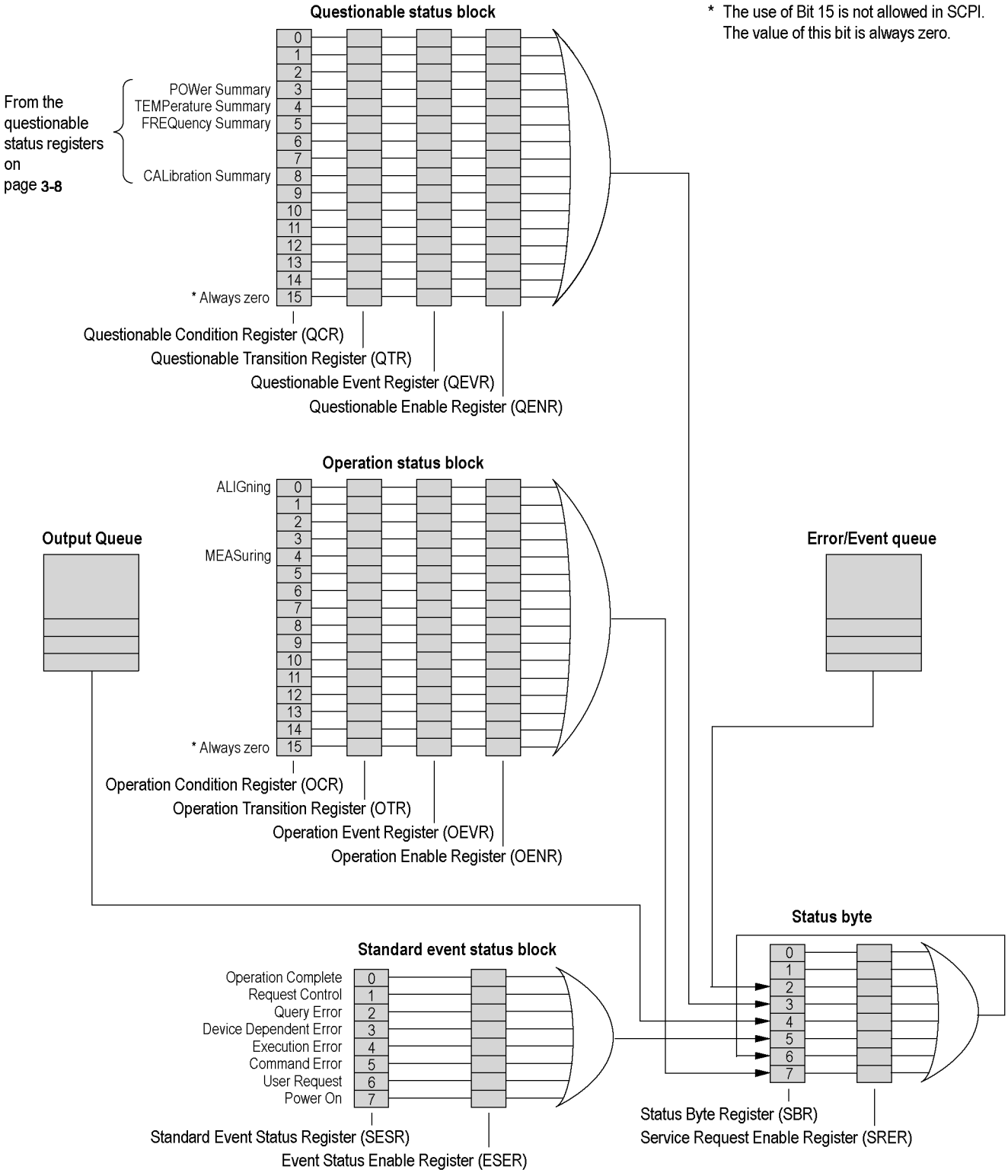


Figure 3-2: Status/Event reporting mechanism (Cont.)

Status Byte

The Status Byte contains the following two registers

- Status Byte Register (SBR)
- Service Request Enable Register (SRER)

Status Byte Register (SBR)

The SBR is made up of 8 bits. Bits 4, 5 and 6 are defined in accordance with IEEE Std 488.2-1987. These bits are used to monitor the output queue, SESR and service requests, respectively. The contents of this register are returned when the *STB? query is used.

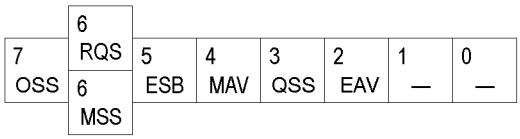


Figure 3-3: Status Byte Register (SBR)

Table 3-2: SBR bit functions

Bit	Description
7	Operation Summary Status (OSS). Summary of the operation status register.
6	Request Service (RQS)/Master Status Summary (MSS). When the instrument is accessed using the GPIB serial poll command, this bit is called the Request Service (RQS) bit and indicates to the controller that a service request has occurred (in other words, that the GPIB bus SRQ line is LOW). The RQS bit is cleared when serial poll ends. When the instrument is accessed using the *STB? query, this bit is called the Master Status Summary (MSS) bit and indicates that the instrument has issued a service request for one or more reasons. The MSS bit is never cleared to 0 by the *STB? query.
5	Event Status Bit (ESB). This bit indicates whether or not a new event has occurred after the previous Standard Event Status Register (SESR) has been cleared or after an event readout has been performed.
4	Message Available Bit (MAV). This bit indicates that a message has been placed in the output queue and can be retrieved.
3	Questionable Summary Status (QSS). Summary of the Questionable Status Byte register.
2	Event Quantity Available (EAV). Summary of the Error Event Queue.
1-0	Not used

Service Request Enable Register (SRER)

The SRER is made up of bits defined exactly the same as bits 0 through 7 in the SBR as shown in the following figure. This register is used by the user to determine what events will generate service requests.

The SRER bit 6 cannot be set. Also, the RQS is not maskable.

The generation of a service request with the GPIB interface involves changing the SRQ line to LOW and making a service request to the controller. The result is that a status byte for which an RQS has been set is returned in response to serial polling by the controller.

Use the *SRE command to set the bits of the SRER. Use the *SRE? query to read the contents of the SRER. Bit 6 must normally be set to 0.

7	6	5	4	3	2	1	0
OSB	—	ESB	MAV	QSB	—	—	—

Figure 3-4: Service Request Enable Register (SRER)

Standard Event Status Block

Reports the power on/off state, command errors, and the running state. It consists of the following registers

- Standard Event Status Register (SESR)
- Event Status Enable Register (ESER)

These registers are made up of the same bits defined in the following figure and table. Use the *ESR? query to read the contents of the SESR. Use the *ESE() command to access the ESER.

7	6	5	4	3	2	1	0
PON	—	CME	EXE	DDE	QYE	—	OPC

Figure 3-5: Standard event status register

Table 3-3: Standard event status register bit definition

Bit	Description
7	Power On (PON). Indicates that the power to the instrument is on.
6	Not used.
5	Command Error (CME). Indicates that a command error has occurred while parsing by the command parser was in progress.
4	Execution Error (EXE). Indicates that an error occurred during the execution of a command. Execution errors occur for one of the following reasons ■ When a value designated in the argument is outside the allowable range of the instrument, or is in conflict with the capabilities of the instrument. ■ When the command could not be executed properly because the conditions for execution differed from those essentially required.
3	Device-Dependent Error (DDE). An instrument error has been detected.

Table 3-3: Standard event status register bit definition (cont.)

Bit	Description
2	Query Error (QYE). Indicates that a query error has been detected by the output queue controller. Query errors occur for one of the following reasons ■ An attempt was made to retrieve messages from the output queue, despite the fact that the output queue is empty or in pending status. ■ The output queue messages have been cleared despite the fact that they have not been retrieved.
1	Not used.
0	Operation Complete (OPC). This bit is set with the results of the execution of the *OPC command. It indicates that all pending operations have been completed.

When an event occurs, the SESR bit corresponding to the event is set, resulting in the event being stacked in the Error/Event Queue. The SBR OAV bit is also set. If the bit corresponding to the event has also been set in the ESER, the SBR ESB bit is also set. When a message is sent to the Output Queue, the SBR MAV bit is set.

Operation Status Block

The operation status block contains conditions that are part of the instrument's normal operation. It consists of the following registers

- Operation Condition Register (OCR)
- Operation Positive/ Negative Transition Register (OPTR/ONTR)
- Operation Event Register (OEVR)
- Operation Enable Register (OENR)

These registers are made up of the same bits defined in the following table and figure. Use the STATus:OPERation commands to access the operation status register set.

15	14	13	12	11	10	9	8	7	6	5	4 MEAS	3	2	1	0 ALIG

Figure 3-6: Operation status register

Table 3-4: Operation status register bit definition

Bit	Description
15	Always zero (0).
14 - 5	Not used.

Table 3-4: Operation status register bit definition (cont.)

Bit	Description
4	Measuring (MEAS). Indicates that the instrument is actively measuring. When the measurement ends after this bit is set in measurement, it is reset. "In measurement" means that one of the following commands is in execution: ■ INITiate commands ■ READ commands
3 - 1	Not used.
0	Aligning (ALIG). Indicates that the instrument is currently performing an alignment. When the alignment ends after this bit is set in alignment, it is reset.

When the specified state changes in the OCR, its bit is set or reset. This change is filtered with a transition register, and the corresponding bit of the OEVR is set. If the bit corresponding to the event has also been set in the OENR, the SBR OSS bit is also set.

Questionable Status Block

The questionable status register set contains bits which give an indication of the quality of various aspects of the signal together with the fanned out registers as described in the next subsections. It consists of the following registers

- Questionable Condition Register (QCR)
- Questionable Positive/Negative Transition Register (QPTR/QNTR)
- Questionable Event Register (QEVR)
- Questionable Enable Register (QENR)

These registers are made up of the same bits defined in the following table and figure. Use the STATus:QUEStionable commands to access the questionable status register set.

15	14 CW	13	12	11	10	9	8 CAL	7	6	5 FREQ	4 TEMP	3 POW	2	1	0

Figure 3-7: Questionable status register

Table 3-5: Questionable status register bit definition

Bit	Description
15	Always zero (0).
14	Command Warning (CW). Indicates a non-fatal warning that relates to the instrument's interpretation of a command, query, or one or more parameters of a specific command or query.
13 - 9	Not used.

Table 3-5: Questionable status register bit definition (cont.)

Bit	Description
8	CALibration Summary (CAL). Summary of the Questionable Calibration register.
7, 6	Not used.
5	FREQuency Summary (FREQ). Summary of the Questionable Frequency register.
4	TEMPerature Summary (TEMP). Summary of the Questionable Temperature register.
3	POWer Summary (POW). Summary of the Questionable Power register.
2 - 0	Not used.

When the specified state changes in the QCR, its bit is set or reset. This change is filtered with a transition register, and the corresponding bit of the QEVR is set. If the bit corresponding to the event has also been set in the QENR, the SBR QSS bit is also set.

Questionable Power Register Set

Refines the power error for the POWer bit in the QCR.

Currently not used.

Questionable Temperature Register Set

Refines the temperature error for the TEMPerature bit in the QCR.

Currently not used.

Questionable Frequency Register Set

The questionable frequency register set is made up of bits defined in the following table and figure. It refines the frequency error for the FREQuency bit in the QCR. Use the STATus:QUEStionable:FREQuency commands to access the questionable frequency register set.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
					ERNV	LTER									

Figure 3-8: Questionable frequency status register

Table 3-6: Questionable frequency status register bit definition

Bit	Description
15	Always zero (0).
14 - 11	Not used.
10	External Ref Not Valid (ERNV). Indicates that the external reference signal is not valid so the instrument is no longer locked to it.
9	Locked To External Ref (LTEF). Indicates that the instrument is locked to the external reference signal.
8 - 0	Not used.

Questionable Calibration Register Set

The questionable calibration register set is made up of bits defined in the following table and figure. It refines the calibration error for the CALibration bit in the QCR. Use the STATus:QUEStionable:CALibration commands to access the questionable calibration register set.

15	14 ALN	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Figure 3-9: Questionable calibration status register

Table 3-7: Questionable calibration status register bit definition

Bit	Description
15	Always zero.
14	Alignment Needed (ALN). Indicates the instrument needs the alignment.
13 - 0	Not used.

Queues

There are two types of queues in the status reporting system used in the analyzer: output queues and event queues.

Output Queue

The output queue is a FIFO (first in, first out) queue and holds response messages to queries, where they await retrieval. When there are messages in the queue, the SBR MAV bit is set.

The output queue will be emptied each time a command or query is received, so the controller must read the output queue before the next command or query is issued. If this is not done, an error will occur and the output queue will be emptied; however, the operation will proceed even if an error occurs.

Event Queue

The event queue is a FIFO queue and stores events as they occur in the analyzer. If more than 32 events occur, event 32 will be replaced with event code -350 ("Queue Overflow"). The error code and text are retrieved using the SYSTem:ERRor queries.

Status and Event Processing Sequence

The following figure shows an outline of the sequence for status and event processing.

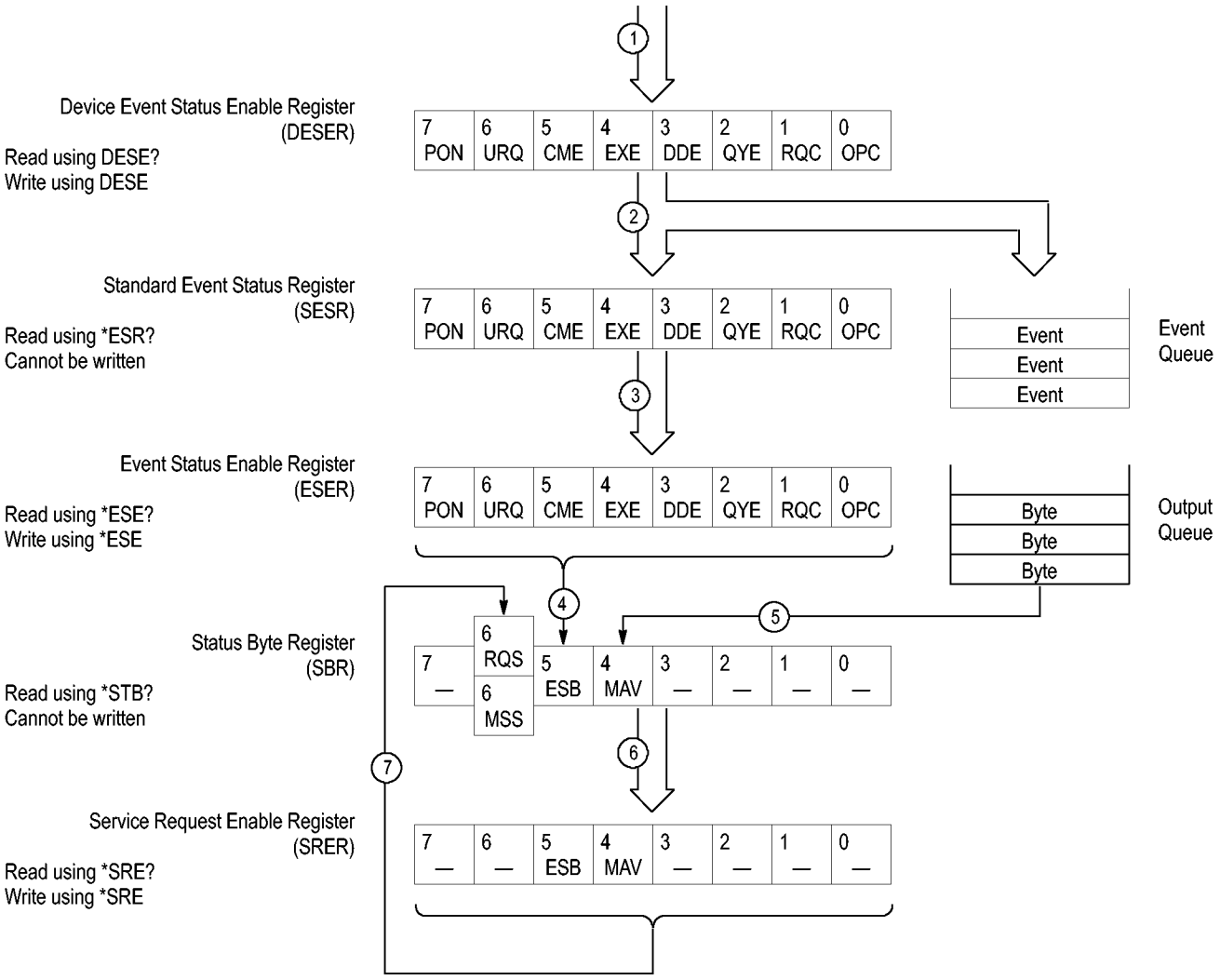


Figure 3-10: Status and event processing sequence

1. If an event has occurred, the SESR bit corresponding to that event is set and the event is placed in the event queue.
2. A bit corresponding to that event in the ESER has is set.
3. The SBR ESB bit is set to reflect the status of the ESER.
4. When a message is sent to the output queue, the SBR MAV bit is set.
5. Setting either the ESB or MAV bits in the SBR sets the respective bit in the SRER.
6. When the SRER bit is set, the SBR MSS bit is set and a service request is generated when using the GPIB interface.

Synchronizing Execution

Almost all commands are executed in the order in which they are sent from the controller, and the execution of each command is completed in a short period of time. However, the following commands perform data analysis in another thread, and another command can thus be executed concurrently

- INITiate commands
- READ commands
- [SENSe]:REANalyze

These commands are designed so that the next command to be sent is executed without waiting for the previous command to be completed. In some cases, a process executed by another command must first be completed before these commands can be executed; in other cases, these commands must be completed before the next command is executed.

You have two options to achieve command synchronization

- Using the status and event reporting function
- Using synchronizing commands

Using the Status and Event Reporting Function

In the following example, a READ command is used to obtain the measurement results while the Operation Condition Register (OCR) is being used to provide synchronization.

```
STATus:OPERation:NTRansition 16
    // Set the filter of the OCR MEASuring bit
STATus:OPERation:ENABle 16
    // Enable the filter of the OCR MEASuring bit
*SRE 128
    // Set the SRER OSS bit
READ:SPECtrum:TRACe
    // Obtain the measurement results
```

The command waits for generation of SRQ.

Using Synchronizing Commands

The IEEE-488.2 common commands include the following synchronizing commands

- *OPC
- *OPC?
- *WAI

Using the *OPC command. The *OPC command sets the SESR OPC bit when all the operations for which it is waiting are completed. If the GPIB interface is in

use, you can synchronize the execution by using this command together with the serial polling or service request function.

The following is a command sequence example:

```
*ESE 1
    // Enable the ESER OPC bit
*SRE 32
    // Enable the SRER ESB bit
ABORt;INITiate:IMMediate;*OPC
    // Wait for SRQ to provide synchronization
```

Using the *OPC? query. The query *OPC? writes ASCII code "1" into the Output Queue when all operations for which it is waiting are completed. You can provide synchronization using the command string as the following example:

```
ABORt;INITiate:IMMediate;*OPC
```

The command waits until "1" is written into the Output Queue. When the command goes to the Output Queue to read the data, a time-out may occur before the data is written into the queue.

Using the *WAI Command. After the process of the preceding command is completed, the *WAI command begins to execute the process of the next command as the following example:

```
ABORt;INITiate:IMMediate;*WAI
    // Wait for the *WAI process to provide synchronization
```

Error Messages and Codes

Error codes with a negative value are SCPI standard error codes; errors with a positive value are unique to the RSA Series Signal Analyzers.

Event codes and messages can be obtained by using the queries `SYSTem:ERRor?` and `SYSTem:ERRor:ALL?` These are returned in the following format

```
<event_code>,"<event_message>"
```

Command Errors

Command errors are returned when there is a syntax error in the command.

Table 3-8: Command errors

Error code	Error message
-100	Command error
-101	Invalid character
-102	Syntax error
-103	Invalid separator
-104	Data type error
-105	GET not allowed
107	Can't replay - settings not compatible with data record
-109	Missing parameter
-111	Header separator error
-112	Program mnemonic too long
-113	Undefined header
-114	Header suffix out of range
-120	Numeric data error
-121	Character
-123	Exponent too large
-124	Too many digits
-128	Numeric data not allowed
-130	Suffix error
-131	Invalid suffix
-134	Suffix too long
-138	Suffix not allowed
-140	Character data error
-141	Invalid character data
-144	Character data too long
-148	Character data not allowed

Table 3-8: Command errors (cont.)

Error code	Error message
-150	String data error
-151	Invalid string data
-158	String data not allowed
-160	Block data error
-161	Invalid block data
-168	Block data not allowed
-170	Command expression error
-171	Invalid expression
-178	Expression data not allowed
-180	Macro error
-181	Invalid outside macro definition
-183	Invalid inside macro definition
-184	Macro parameter error

Execution Errors

These error codes are returned when an error is detected while a command is being executed.

Table 3-9: Execution errors

Error code	Error message
-200	Execution error
-201	Invalid while in local
-202	Settings lost due to RTL
-210	Trigger error
-211	Trigger ignored
-212	Arm ignored
-213	Init ignored
-214	Trigger deadlock
-215	Arm deadlock
-220	Parameter error
-221	Settings conflict
-222	Data out of range
-223	Too much data
-224	Illegal parameter value
-225	Out of memory
-226	Lists not same length

Table 3-9: Execution errors (cont.)

Error code	Error message
-230	Data corrupt or stale
-231	Data questionable
-240	Hardware error
-241	Hardware missing
-250	Mass storage error
-251	Missing mass storage
-252	Missing media
-253	Corrupt media
-254	Media full
-255	Directory full
-256	Filename not found
-257	Filename error
-258	Media protected
-260	Execution expression error
-261	Math error in expression
-270	Execution macro error
-271	Macro syntax error
-272	Macro execution error
-273	Illegal macro label
-274	Execution macro parameter error
-275	Macro definition too long
-276	Macro recursion error
-277	Macro redefinition not allowed
-278	Macro header not found
-280	Program error
-281	Cannot create program
-282	Illegal program name
-283	Illegal variable name
-284	Program currently running
-285	Program syntax error
-286	Program runtime error

Device Specific Errors

These error codes are returned when an internal instrument error is detected. This type of error may indicate a hardware problem.

Table 3-10: Device specific errors

Error code	Error message
-300	Device specific error
-310	System error
-311	Memory error
-312	PUD memory lost
-313	Calibration memory lost
-314	Save/Recall memory lost
-315	Configuration memory lost
-330	Self test failed
-350	Queue overflow

Query Errors

These error codes are returned in response to an unanswered query.

Table 3-11: Query errors

Error code	Error message
-400	Query error
-410	Query interrupted
-420	Query unterminated
-430	Query deadlocked
-440	Query unterminated after indefinite period

Device Errors

These error codes are unique to the RSA Series. They are classified into three groups: global, measurement, and source conditions, as shown in the following tables.

Table 3-12: Device errors, global condition

Error code	Error message
100	Setup error
101	Disabled: data is from swept acquisition
102	Disabled: swept settings; Acquire data while display is selected
103	Acquisition bandwidth too small for current setup
104	Can't get acquisition data record
105	Can't open the requested display
106	Analysis failure

Table 3-12: Device errors, global condition (cont.)

Error code	Error message
107	Analysis length was limited
108	Analysis length too small for current setup
109	No math trace: unmatched trace lengths
110	Analysis time was adjusted
111	Not enough samples for current setup
112	Can't replay. Data is from swept acquisition.
113	Can't replay. Live data needed for swept settings.
114	Recall error: setup not completely restored
115	Recall failure: problem with file or file contents
116	Save failure: file not saved
117	Unexpected software error. Please cycle power and try again.
118	Export failure: file not saved
119	Export failure: unable to open results file for export. File not saved.
120	Search condition for this result is already defined.
121	Search condition for this result was not found.
124	Load failed: <filename>
125	Store error: file not saved.
126	No Math trace: unmatched trace X range
127	Not enough memory for measurement
128	Incomplete analysis
129	Not enough samples for current setup
130	Mask creation error: <reason message>

Table 3-13: Device errors, measurement condition

Error code	Error message
1000	TDBW actual (TDBW: Time Domain Bandwidth)
1002	RBW increased to
1003	RBW limited by acquisition bandwidth to
1004	RBW conflict. Increase span or analysis length
1011	Audio disabled: configuration problem
1012	Audio Demod disabled: swept acquisition
1013	Audio Demod disabled: trigger in use
1014	Audio disabled: IF band outside Acq BW
1015	Calibration error. See Windows Event Viewer for error detail.
OBW errors	
1016	Analysis failure: AcqBW < MeasBW + (5 x RBW)
1017	Analysis failure: AcqBW must be 10 kHz or more

Table 3-13: Device errors, measurement condition (cont.)

Error code	Error message
1018	x dB BW > Meas BW
Pulse errors	
1019	AcqBW too low for current Chirp BW setting
1020	Impulse Response only valid for Chirp modulation
1023	Not enough memory for measurement
Other measurements	
1024	BW actual (limited by Acq BW)
1025	CISPR not available in FastFrame. Uncheck FastFrame in the Acquire panel.
1026	Analysis length must be in auto.
1027	Carrier not found
1029	CISPR accuracy limited by acq memory. Adjust RBW or freq range.
1030	CISPR: Acq BW too small for RBW. Try increasing span or freq range.
1031	Insufficient data for CISPR. Acquire while display is selected.
1032	VBW increased - Analysis Length too short
1033	VBW does not use full Analysis Length.
1034	Analysis time was adjusted\nReached sample limit of

Table 3-14: Device errors, source condition

Error code	Error message
2008	Not calibrated
2009	FMT RBW limited by Acq BW to <XX>
2010	FMT RBW increased by Acq BW to <XX>
2011	Hardware failure - see Windows Event Viewer
2012	FMT disabled: Acq BW too small for current setup
2013	Hardware failure detected by diagnostics
2014	FMT disabled: RefLev too low for Volts/Watts units
2015	FMT disabled: settings conflict with selected measurement
2016	Preamp freq range exceeded
2024	RBW limited by Span to <XX>
2025	RBW increased by Span to <XX>
2028	External frequency reference signal not valid. Using internal reference.
2029	Unable to lock to external frequency reference. Using internal reference.
2033	Alignment error. Please run alignment again. If the problem persists, contact your Tektronix Service Center.
2034	Operational error: unable to complete operation. Please try again. If the problem persists, contact your Tektronix Service Center. See Windows Event Viewer for error detail.

Table 3-14: Device errors, source condition (cont.)

Error code	Error message
2035	Hardware error: unable to configure hardware. Please try again. If the problem persists, contact your Tektronix Service Center. See Windows Event Viewer for error detail.
2036	Shutting down - internal temperature is too high. Check fans and airflow. If the problem persists, contact your Tektronix Service Center.
2037	Hardware error detected. To clear error, exit and restart the application. If the problem persists, contact your Tektronix Service Center.
2038	Disabled: RefLev too low\nfor Volts/Watts units
2042	Ext Corr > 20 dB pk-pk in acq segment
2044	Combined External Correction tables exceed the 60 dB peak-to-peak limit. External Correction tables were disabled. Please check table values and try again.
2045	Disabled: FastFrame doesn't support swept settings.
2046	Attenuator use currently exceeds the cautionary limit of 1200 changes per hour. Operation was stopped to protect against premature wear-out. The monitor function can be temporarily disabled in the Amplitude control panel or over the programmatic interface. Specified lifetime for an attenuator is 10,000,000 changes.
2047	Disabled: settings conflict with selected measurement
2048	Done saving — restart with Stop, then Run. The maximum number of files saved per run have been saved. To restart another FastSave acquisition, send INITIATE.

Status Conditions

Status conditions do not appear in the event queue, they are only returned by a :STATus:<measurement>EVENts? query.

Global Status

Global status conditions can be returned with any :STATus:<measurement>EVENts? query.

Table 3-15: Global status conditions

Status code	Status message
10107	Analysis Length was limited
10108	Analysis length too small for current setup
10110	Analysis time was adjusted
10122	Search result not enabled
10123	Search result no longer available

Measurement Status

Measurement status conditions are unique to specific measurement groups.

Table 3-16: Measurement status conditions

Status code	Status message
Pulse Measurements	
11001	Avg Tx only available in power units
11005	Pulse detection error
11006	Setup error: Define > Measurement Point is too late
11007	No pulses found
11008	No FFT (not all pulses have results)
11020	No FFT (not enough pulses)
11022	Analysis time was adjusted - reached time limit
11039	No sidelobe found
GP Digital Modulation	
11009	No burst detected
11034	Display not valid for selected modulation type
11035	Samples per symbol > 200. Decrease Measurement BW
11036	Too many samples per symbol. Decrease Measurement BW

Table 3-16: Measurement status conditions (cont.)

Status code	Status message
Chan Power and ACPR	
11010	Measuring Noise
Other	
11028	CISPR: (progress message)
11037	Ext corrections enabled for traces
Settling Time	
11041	Signal not settled
11042	Settled duration too short
11043	First transition not found
11044	Meas Length limited by Analysis Length to xx.x xs
11045	Not enough samples: increase Meas BW or Meas Length
11046	Measurement Length exceeds limit of xxM points

Source Status

Source status conditions can be returned with any :STATus:<measurement>EVENts? query.

Table 3-17: Source status conditions

Status code	Status message
12000	Data acquired during RF ADC overrange
12005	Aligning
12006	Not aligned
12007	Unaligned data
12009	Data from uncalibrated instrument
12010	Dither: manual control
12012	Data acquired during hardware failure
12014	Data acquired during RF dig gain overflow
12015	Locking to ext freq ref signal...
12018	RF Attenuator: manual control
12019	Saving acquisition
12020	Recalling acquisition data
12022	Disabled: data is from swept acquisition
12023	Disabled: Freq Mask Trigger in use
12026	Acq Sampling params: manual control
12027	Swept: RF Trig invalid for most signals
12030	Data acquired during RF ADC overrange

Table 3-17: Source status conditions (cont.)

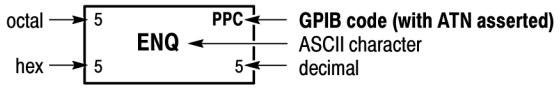
Status code	Status message
12031	Data acquired during RF dig gain overflow
12039	Decrease RefLev or increase\nAtten to prevent data errors
12040	Possible data errors - acquired with RefLev too much higher than RFAtten
12043	Ext Corr error - tables not applied
12056	Warming up
12057	Data acquired during warm-up
12063	Triggers disabled when DPX is swept
12064	Waiting for trigger. Open the DPX display.
12066	Data exceeds preamp range

Appendices

Appendix A: Character Charts

B7 B6 B5 BITS B4 B3 B2 B1	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
	CONTROL		NUMBERS SYMBOLS		UPPER CASE		LOWER CASE	
0 0 0 0	0 NUL 0 0	20 DLE 10 16	40 LA0 SP 20 32	60 LA16 0 30 48	100 TA0 @ 40 64	120 TA16 P 50 80	140 SA0 ` 60 96	160 SA16 p 70 112
0 0 0 1	1 GTL SOH 1 1	21 LL0 DC1 11 17	41 LA1 ! 21 33	61 LA17 1 31 49	101 TA1 A 41 65	121 TA17 Q 51 81	141 SA1 a 61 97	161 SA17 q 71 113
0 0 1 0	2 STX 2 2	22 DC2 12 18	42 LA2 " 22 34	62 LA18 2 32 50	102 TA2 B 42 66	122 TA18 R 52 82	142 SA2 b 62 98	162 SA18 r 72 114
0 0 1 1	3 ETX 3 3	23 DC3 13 19	43 LA3 # 23 35	63 LA19 3 33 51	103 TA3 C 43 67	123 TA19 S 53 83	143 SA3 c 63 99	163 SA19 s 73 115
0 1 0 0	4 SDC EOT 4 4	24 DCL DC4 14 20	44 LA4 $ 24 36	64 LA20 4 34 52	104 TA4 D 44 68	124 TA20 T 54 84	144 SA4 d 64 100	164 SA20 t 74 116
0 1 0 1	5 PPC ENQ 5 5	25 PPU NAK 15 21	45 LA5 % 25 37	65 LA21 5 35 53	105 TA5 E 45 69	125 TA21 U 55 85	145 SA5 e 65 101	165 SA21 u 75 117
0 1 1 0	6 ACK 6 6	26 SYN 16 22	46 LA6 & 26 38	66 LA22 6 36 54	106 TA6 F 46 70	126 TA22 V 56 86	146 SA6 f 66 102	166 SA22 v 76 118
0 1 1 1	7 BEL 7 7	27 ETB 17 23	47 LA7 ' 27 39	67 LA23 7 37 55	107 TA7 G 47 71	127 TA23 W 57 87	147 SA7 g 67 103	167 SA23 w 77 119
1 0 0 0	10 GET BS 8 8	30 SPE CAN 18 24	50 LA8 (28 40	70 LA24 8 38 56	110 TA8 H 48 72	130 TA24 X 58 88	150 SA8 h 68 104	170 SA24 x 78 120
1 0 0 1	11 TCT HT 9 9	31 SPD EM 19 25	51 LA9) 29 41	71 LA25 9 39 57	111 TA9 I 49 73	131 TA25 Y 59 89	151 SA9 i 69 105	171 SA25 y 79 121
1 0 1 0	12 LF A 10	32 SUB 1A 26	52 LA10 * 2A 42	72 LA26 : 3A 58	112 TA10 J 4A 74	132 TA26 Z 5A 90	152 SA10 j 6A 106	172 SA26 z 7A 122
1 0 1 1	13 VT B 11	33 ESC 1B 27	53 LA11 + 2B 43	73 LA27 ; 3B 59	113 TA11 K 4B 75	133 TA27 [5B 91	153 SA11 k 6B 107	173 SA27 { 7B 123
1 1 0 0	14 FF C 12	34 FS 1C 28	54 LA12 , 2C 44	74 LA28 < 3C 60	114 TA12 L 4C 76	134 TA28 \ 5C 92	154 SA12 l 6C 108	174 SA28 ¦ 7C 124
1 1 0 1	15 CR D 13	35 GS 1D 29	55 LA13 – 2D 45	75 LA29 = 3D 61	115 TA13 M 4D 77	135 TA29] 5D 93	155 SA13 m 6D 109	175 SA29 } 7D 125
1 1 1 0	16 SO E 14	36 RS 1E 30	56 LA14 . 2E 46	76 LA30 > 3E 62	116 TA14 N 4E 78	136 TA30 ^ 5E 94	156 SA14 n 6E 110	176 SA30 ~ 7E 126
1 1 1 1	17 SI F 15	37 US 1F 31	57 LA15 / 2F 47	77 UNL ? 3F 63	117 TA15 O 4F 79	137 UNT – 5F 95	157 SA15 o 6F 111	177 RUBOUT (DEL) 7F 127
	ADDRESSED COMMANDS	UNIVERSAL COMMANDS	LISTEN ADDRESSES		TALK ADDRESSES		SECONDARY ADDRESSES OR COMMANDS	

KEY



Tektronix
REF: ANSI STD X3.4-1977
IEEE STD 488.1-1987
ISO STD 646-2973

Appendix B: GPIB Interface Specification

This appendix lists and describes the GPIB functions and messages the instrument can implement.

Interface Functions

The following table lists the GPIB interface functions this instrument implements. Each function is briefly described.

Table B-1: GPIB interface function implementation

Interface function	Implemented subset	Capability
Source Handshake (SH)	SH1	Complete
Acceptor Handshake (AH)	AH1	Complete
Talker (T)	T6	Basic Talker, Serial Poll Unaddress if my-listen-address (MLA) No Talk Only mode
Listener (L)	L4	Basic Listener Unaddress if my talk address (MTA) No Listen Only mode
Service Request (SR)	SR1	Complete
Remote/Local (RL)	RL0	None
Parallel Poll (PP)	PP0	None
Device Clear (DC)	DC1	Complete
Device Trigger (DT)	DT0	None
Controller (C)	C0	None
Electrical Interface	E2	Three-state driver

- Source Handshake (SH). Enables a talking device to support the coordination of data transfer. The SH function controls the initiation and termination of data byte transfers.
- Acceptor Handshake (AH). Enables a listening device to coordinate data reception. The AH function delays data transfer initiation or termination until the listening device is ready to receive the next data byte.
- Talker (T). Enables a device to send device-dependent data over the interface. This capability is available only when the device is addressed to talk, and uses a one-byte address.
- Listener (L). Enables a device to receive device-dependent data over the interface. This capability is available only when the device is addressed to listen, and uses a one-byte address.
- Service Request (SR). Enables a device to assert an SRQ (Service Request) line to notify the controller when it requires service.
- Remote/Local (RL). Enables a device to respond to both the GTL (Go To Local) and LLO (Local Lock Out) interface messages.
- Parallel Poll (PP). Enables a device to respond to the following interface messages: PPC, PPD, PPE, and PPU, as well as to send out a status message when the ATN (Attention) and EOI (End or Identify) lines are asserted simultaneously.
- Device Clear (DC). Enables a device to be cleared or initialized, either individually, or as part of a group of devices.
- Device Trigger (DT). Enables a device to respond to the GET (Group Execute Trigger) interface message when acting as a listener.
- Controller (C). Enables a device that has this capability to send its address, universal commands, and addressed commands to other devices over the interface.
- Electrical Interface (E). Identifies the electrical interface driver type. The notation E1 means the electrical interface uses open collector drivers, E2 means the electrical interface uses three-state drivers.

Interface Messages

Table B-2: Standard interface messages

Message	Type [1]	Implemented
Device Clear (DCL)	UC	Yes
Local Lockout (LLO)	UC	No
Serial Poll Disable (SPD)	UC	Yes
Serial Poll Enable (SPE)	UC	Yes

Table B-2: Standard interface messages (cont.)

Message	Type [1]	Implemented
Parallel Poll Unconfigure (PPU)	UC	No
Go To Local (GTL)	AC	Yes
Selected Device Clear (SDC)	AC	Yes
Group Execute Trigger (GET)	AC	No
Take Control (TCT)	AC	No
Parallel Poll Configure (PPC)	AC	No

1 UC: Universal command; AC: Address command

- Device Clear (DCL). Will clear (initialize) all devices on the bus that have a device clear function, whether or not the controller has addressed them.
- Local Lockout (LLO). Disables the return to local function.
- Serial Poll Disable (SPD). Changes all devices on the bus from the serial poll state to the normal operating state.
- Serial Poll Enable (SPE). Puts all bus devices that have a service request function into the serial poll enabled state. In this state, each device sends the controller its status byte, instead of its normal output, after the device receives its talk address on the data lines. This function may be used to determine which device sent a service request.
- Go To Local (GTL). Causes the listen-addressed device to switch from remote to local (front-panel) control.
- Select Device Clear (SDC). Clears or initializes all listen-addressed devices.
- Group Execute Trigger (GET). Triggers all applicable devices and causes them to initiate their programmed actions.
- Take Control (TCT). Allows the controller in charge to pass control of the bus to another controller on the bus.
- Parallel Poll Configure (PPC). Causes the listen-addressed device to respond to the secondary commands Parallel Poll Enable (PPE) and Parallel Poll Disable (PPD), which are placed on the bus following the PPC command. PPE enables a device with parallel poll capability to respond on a particular data line. PPD disables the device from responding to the parallel poll.

Appendix C: SCPI Conformance Information

All commands for the RSA Series analyzers are based on SCPI Version 1999.0. The following table lists the commands that are defined in the SCPI 1999.0 Standard. The other commands not listed in the table are not defined in the SCPI 1999.0 Standard.

Table C-1: SCPI 1999.0-defined commands

Command group	Command		
IEEE common	*CAL		
	*CLS		
	*ESE		
	*ESR		
	*IDN		
	*OPC		
	*OPT		
	*RST		
	*SRE		
	*STB		
	*TRG		
	*WAI		
ABORt	:ABORt		
INITiate	:INITiate	:CONTinuous	
		[:IMMediate]	
STATus	:STATus	:OPERation	:CONDition?
			:ENABle
			[:EVENt]?
			:NTRansition
			:PTRansition
		:PRESet	
		:QUEStionable	:CONDition?
			:ENABle
			[:EVENt]?
			:NTRansition
			:PTRansition

Table C-1: SCPI 1999.0-defined commands (cont.)

Command group	Command				
SYSTem	:SYSTem	:COMMunicate	:GPIB	[:SELF]	:ADDRess
		:DATE			
		:ERRor	:ALL?		
			:CODE	:ALL?	
				[:NEXT]?	
			:COUNt?		
			[:NEXT]?		
		:PRESet			
		:TIME			
		:VERSion?			
UNIT	:UNIT	:POWer			

Appendix D: Creating a DPX Frame Access Plugin

A DPX Frame Access plugin gives access to DPX frames (bitmap and traces) from within the RSA process. The frame is accessed directly from its storage in memory, and so this method provides faster access to this data than from the ASCII (GPIB) interface.

A DPX Frame Access plugin consists of a managed assembly (dll) that contains a class that implements the IDPXFrameAccess interface. See below:

```
/// <summary>
/// If an external assembly is found that contains a type
/// that implements this interface, then it will be given
/// access to DPX frame data synchronous with data
/// coming from the hardware.
/// </summary>
public interface IDPXFrameAccess :  IDisposable
{
   /// <summary>
   /// A frame is available.  It will get called for
   /// every stitch.
   /// </summary>
   /// <param name="frame">valid only during the duration
   /// of the function call </param>
   void FrameAvailable(IDPXFrame frame);
}
```

The object that is passed into the FrameAvailable function is accessed via its IDPXFrame interface. See below:

```
/// <summary>
/// A DPX frame
///
/// </summary>
public interface IDPXFrame
{
   /// <summary>
   /// Start frequency for traces and bitmap, Hz
   /// </summary>
   double StartFrequency { get; }
   /// <summary>
   /// Stop frequency for traces and bitmap, Hz
   /// </summary>
   double StopFrequency { get; }
   /// <summary>
   /// Top value of graph - current units
   /// </summary>
   double TopOfGraph { get; }
   /// <summary>
```

```
/// Bottom value of graph - current units
/// </summary>
double BottomOfGraph { get; }
/// <summary>
/// What type of frame?
/// </summary>
DPXFrameType FrameType { get; }
/// <summary>
/// This is true if the frame is completed.
/// It will be false if swept, and not all
/// of the intermediate stitches have been
/// completed.  It will be true when all stitches
/// have been completed, or if real-time.
/// </summary>
bool FrameComplete { get; }

/// <summary>
/// Get dimensions for bitmap
/// </summary>
/// <param name="rowCount"></param>
/// <param name="columnCount"></param>
void GetDimensions(out int rowCount, out int
columnCount);

/// <summary>
/// <param name="x"></param>
void GetReferenceToBitmap(out ushort[] x);

/// <summary>
/// Get reference to array holding bitmap.
/// Do not change the values in this array
/// </summary>
/// <param name="x"></param>
void GetReferenceToBitmap(out float[] x);

/// <summary>
/// Get copy of Trace1.  This will be the Max trace
/// </summary>
/// <param name="yValues">function will allocate
/// the array, as necessary</param>
void GetTrace1(ref float[] yValues);

/// <summary>
/// Get copy of Trace2.  This will be the Min trace
/// </summary>
/// <param name="yValues">function will allocate
/// the array, as necessary</param>
void GetTrace2(ref float[] yValues);

/// <summary>
```

```
    /// Get copy of Trace3.  This will be the Avg trace
    /// </summary>
    /// <param name="yValues">function will allocate
    /// the array, as necessary</param>
    void GetTrace3(ref float[] yValues);
}

public enum DPXFrameType
{

    /// <summary>
    Standard,
    /// </summary>
    Enhanced,
}
```

When the DPX measurement is running within the RSA application, the IDPXFrameAccess.FrameAvailable function is called once for every frame that becomes available from the DPX processing hardware. When the DPX measurement is operating in Real Time mode, FrameAvailable will be called for every Real Time frame. When the DPX measurement is operating in Swept mode, the total frequency span is covered by stitching together multiple real time spans at increasing center frequencies. In this case, FrameAvailable will be called for every stitch, with IDPXFrame.FrameComplete set to true on the last stitch.

The DPXFrameAccess assembly should not use the reference to the IDPXFrame object outside of the FrameAvailable function. It is only valid within the context of the FrameAvailable function call.

The creator of a DPXFrameAccess plugin should refer to the information in this programmer manual about the format of bitmap and trace data returned in the IDPXFrame object. Specifically, the *FETCh:DPX:TRACe:xxx?* queries are applicable. Traces that are not enabled are not computed and will not be available in the IDPXFrame.

Cautions about processing time within FrameAvailable function – in Real Time mode, DPX frames become available at a constant rate. As more time is taken in the FrameAvailable function, less CPU time is available for processing RF acquisitions (all the other measurements). If too much time is taken in the FrameAvailable function, frames may be missed and the update rate of other measurements may significantly slow down as they operate at lower priority than the DPX measurement.

To create a DPX Frame Access plugin, start with the example file attached to the PDF version of this manual. It has a class that implements IDPXFrameAccess and gives examples of accessing the various properties and methods of IDPXFrame.

NOTE. *An example text file, DPXFrameAccessExample.cs, is attached to this PDF file. To access the attached file, open the PDF file and click on the paperclip icon on the lower-left side of the document viewer.*

Note that the build of this assembly will need to reference the MeasurementsShared.dll assembly in the C:\Program Files\Tektronix\XXXX\RSA\Client directory, where XXXX is either RSA5100B, or SignalVu, depending on the product..

Copy the DPX Frame Access assembly you have created to the C:\Program Files\Tektronix\XXXX\RSA\Plugins\DPX directory (you will need to create the directory). When the DPX measurement is started, it will discover the assembly and the FrameAvailable function will be called.

Glossary

Glossary

ASCII
Acronym for the American Standard Code for Information Interchange. Controllers transmit commands to the instrument using ASCII character encoding.

ASK
Acronym for Amplitude Shift Keying. The process, or result of a process, in which the amplitude of the carrier is varied in accordance with the state of a digital input signal.

BNF (Backus-Naur Form)
A standard notation system for command syntax diagrams. The syntax diagrams in this manual use BNF notation.

Controller
A computer or other device that sends commands to and accepts responses from the analyzer.

EVM (Error Vector Magnitude)
The magnitude of an error of an actual signal relative to an ideal signal in a constellation display.

GPIB
Acronym for General Purpose Interface Bus, the common name for the communications interface system defined in IEEE Std 488.

IEEE
Acronym for the Institute for Electrical and Electronic Engineers.

IS95
Acronym for Interim Standard-95. The standards name for first-generation CDMA cellphone technology.

Modulation
The process of varying some characteristic of a signal with a second signal.

PSK
Acronym for Phase Shift Keying. The process, or result of a process, in which the carrier phase is varied discretely in accordance with a digital code.

QAM
Acronym for Quadrature Amplitude Modulation. The process, or result of a process, in which the amplitude and phase of the carrier are varied concurrently by synthesizing two orthogonal ASK waves (see ASK).

Index

Symbols and Numbers

A

C

D

E

F

I

M

O

R

S

T

U

W