**Programming Manual**

# Remote Spectrum Monitor MS27201A Microwave Spectrum Monitor Module MS27200A

## High-Performance RF Spectrum Monitor
## OEM Assembly for Custom System Integration

**MS2720xA-0709 9 kHz to 9 GHz (Option 709)**
**MS2720xA-0714 9 kHz to 14 GHz (Option 714)**
**MS2720xA-0720 9 kHz to 20 GHz (Option 720)**
**MS2720xA-0726 9 kHz to 26.5 GHz (Option 726)**
**MS2720xA-0732 9 kHz to 32 GHz (Option 732)**
**MS2720xA-0743 9 kHz to 43.5 GHz (Option 743)**
**MS2720xA-0754 9 kHz to 54 GHz (Option 754)**

Anritsu Company
490 Jarvis Drive
Morgan Hill, CA 95037-2809
USA

Part Number: 10580-00481
Revision: H
Published: 2025

## Unauthorized Use or Disclosure

Anritsu Company has prepared the product user documentation for use by Anritsu Company personnel and customers as a guide for the proper installation, operation, and maintenance of Anritsu Company equipment and software programs. The drawings, specifications, and information contained therein are the property of Anritsu Company, and any unauthorized use of these drawings, specifications, and information is prohibited; they shall not be reproduced, copied, or used in whole or in part as the basis for manufacture or sale of the equipment or software programs without the prior written consent of Anritsu Company.

## Export Management

The Anritsu products identified herein and their respective manuals may require an Export License or approval by the government of the product country of origin for re-export from your country. Before you export these products or any of their manuals, please contact Anritsu Company to confirm whether or not these items are export-controlled. When disposing of export-controlled items, the products and manuals must be broken or shredded to such a degree that they cannot be unlawfully used for military purposes.

# Table of Contents

# Chapter 1 — Programming with SCPI

## 1-1 Scope of Manual

This manual provides SCPI programming information and a list of SCPI programming commands for the Anritsu MS27201A Remote Spectrum Monitor (RSM) and MS27200A Microwave Spectrum Monitor Module (MSMM).

### Additional Documentation

| Document Part Number | Description |
|---|---|
| 10100-00064 | Product Information, Compliance, and Safety Notices |
| 11410-02833 | MS27201A Remote Spectrum Monitor Technical Data Sheet |
| 10580-02941 | MS27200A Microwave Spectrum Monitor Module Technical Data Sheet |
| 10580-00480 | MS27201A Remote Spectrum Monitor User Guide |
| 10580-00506 | MS27200A Microwave Spectrum Monitor Module User Guide |

Updates, if any, can be downloaded from the Library tab on the Anritsu product page:

https://www.anritsu.com/en-us/test-measurement/products/ms2720xa

https://www.anritsu.com/en-us/test-measurement/products/ms27200a

## 1-2 Introduction

This chapter provides an introduction to Standard Commands for Programming Instruments (SCPI). The SCPI standard defines a set of programming commands and standards for use by all SCPI compatible devices. SCPI is intended to provide a consistent environment for program development. It does so by defining standardized controller messages, responses, and message formats for all SCPI compatible devices. The following sections include descriptions of the command types, hierarchical command structure, command subsystems, data parameters, and notational conventions with examples.

# 1-3 Remote Programming Setup and Interface

Remote programming and operation of the instrument/module is accomplished via the Ethernet or USBTMC (MS27200A). The following sections provide information about the interface connections, cable requirements, and remote operation setup.

## Network Interface Connection and Setup

The MS2720xA uses Ethernet or USBTMC (MS27200A) to communicate remotely with a controller. Most functions (except power on/off) can be controlled via a network connection to a PC connected directly (with an Ethernet cross-over cable) or through a network. The Anritsu Remote Tool (MS27201A) supports the TCP/IP raw socket network protocol.

| **Note** | The Anritsu Remote Tool is not supported by MS27200A module. System functions, supported measurements and other common features are controlled using SCPI commands only. |
|---|---|

Ethernet networking uses a bus or star topology in which all of the interfacing devices are connected to a central cable called the bus, or are connected to a hub. Ethernet uses *Carrier Sense Multiple Access/Collision Detection* (CSMA/CD) access method to handle simultaneous transmissions over the bus. This standard enables network devices to detect simultaneous data channel usage, called a *collision*, and provides for a contention protocol. When a network device detects a collision, the CSMA/CD standard dictates that the data is retransmitted after waiting a random amount of time. If a second collision is detected, the data is again retransmitted after waiting twice as long. This is known as exponential back off.

The TCP/IP setup requires the following:

- IP Address: Every computer and electronic device in a TCP/IP network requires an IP address. An IPv4 address has four numbers (each between 0 and 255) separated by periods. For example: 128.111.122.42 is a valid IP address.
- Subnet Mask: The subnet mask distinguishes the portion of the IP address that is the network ID from the portion that is the station ID. The subnet mask 255.255.0.0, when applied to the IP address given above, would identify the network ID as 128.111 and the station ID as 122.42. All stations in the same local area network should have the same network ID, but different station IDs.
- Default Gateway: A TCP/IP network can have a gateway to communicate beyond the LAN identified by the network ID. A gateway is a computer or electronic device that is connected to two different networks and can move TCP/IP data from one network to the other. A single LAN that is not connected to another LAN requires a default gateway setting of 0.0.0.0. If you have a gateway, then the default gateway would be set to the appropriate value of your gateway.
- Ethernet Address: An Ethernet address, or *Media Access Control* (MAC) address, is a unique 48-bit value that identifies a network interface card to the rest of the network. Every network card has a unique Ethernet address permanently stored into its memory.
- Remote programming and operation is accomplished via a TCP/IP raw socket connection to port 9001. The Anritsu Remote Tool (MS27201A) must establish a TCP/IP raw socket connection at port 9001 to the instrument.
- The MS2720xA is shipped with default static IP configuration. Refer to "Static IP Configuration" on page 1-4 for detailed information.
- The Anritsu Remote Tool is used to connect the MS27201A to the network through `IP ADDRESS` or to its `HOSTNAME`. If using DHCP instead of a static IP, using the `HOSTNAME` may be more reliable for finding an instrument on a network. Refer to "Dynamic Configuration - Anritsu Remote Tool" on page 1-11 for detailed instructions.
- MS27200A does not support Anritsu Remote Tool, so refer to "Ethernet Configuration" on page 1-4 for information on using SCPI commands to modify the network configuration.
- MS27200A supports USBTMC interface connection (USB TYPE-C). Refer to "Sending SCPI Commands with NI-VISA" on page 1-14 to obtain detailed information.

- You may need to contact your network administrator to ensure network security policies, antivirus, and firewall settings do not block the access to the controlling computer and its ports.

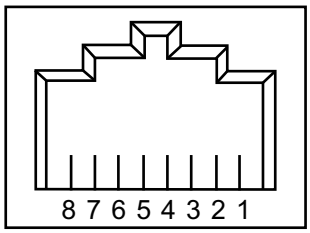
The MS2720xA can be configured for *Dynamic Host Configuration Protocol* (DHCP), an Internet protocol that automates the process of setting IP addresses for devices that use TCP/IP, and is the most common method of configuring a device for network use.

To determine if a network is set up for DHCP, connect the instrument/module to the network and select DHCP protocol. Power cycle the instrument. If the network is set up for DHCP, the assigned IP address should be displayed in the network settings. For details regarding enabling DHCP configuration for MS27200A, refer to MS27200 Microwave Spectrum Monitor Module User Guide (PN:10580-00506).

### Ethernet Connection

Interface between the instrument and other devices on the network is via a category five (CAT-5) interface cable connected to a network. This cable uses four twisted pairs of insulated copper wires terminated into an RJ45 connector. CAT-5 cabling is capable of supporting frequencies up to 100 MHz and data transfer speeds up to 1 Gbps, which accommodates 1000Base-T, 100Base-T, and 10Base-T networks. CAT-5 cables are based on the EIA/TIA 568 Commercial Building Telecommunications Wiring Standard developed by the Electronics Industries Association. A pinout diagram is shown in Table 1-1.

**Table 1-1.** 8-pin Ethernet RJ45 Connector Pinout Diagram




| Pin | Name | Description | Wire Color |
|---|---|---|---|
| 1 | TX+ or BI_DA+ | Transmit or bidirectional data | White/Green |
| 2 | TX– or BI_DA- | Transmit or bidirectional data | Green |
| 3 | RX+ or BI_DB+ | Receive or bidirectional data | White/Orange |
| 4 | BI_DC+ | Not used (common mode termination) or bidirectional | Blue |
| 5 | BI_DC- | Not used (common mode termination) or bidirectional | White/Blue |
| 6 | RX– or BI_DB- | Receive or bidirectional data | Orange |
| 7 | BI_DD+ | Not used (common mode termination) or bidirectional | White/Brown |
| 8 | BI_DD- | Not used (common mode termination) or bidirectional | Brown |

Integrated into the RJ45 connector are two LEDs. The amber LED indicates the presence of LAN voltages (a live LAN connection) while the green LED flashes to show that LAN traffic is present. The instrument IP address and its `HOSTNAME` are set via the System menu (upper left corner) and accessing the ETHERNET settings menu.

TCP/IP connectivity requires setting up the parameters described at the beginning of this section. The following is a brief overview of how to set up a general LAN connection on the MS27201A.

| Note | You may need to consult your network documentation or network administrator for assistance in configuring your network setup. |
|---|---|

## Ethernet Configuration

Most instrument functions (except power on/off) can be controlled via a network connection to a PC connected directly (with an Ethernet cross-over cable) or through a network. The product firmware supports the TCP/IP raw socket network protocol.

### Static IP Configuration

The MS2720xA is shipped with the following default static IP configuration:

- **DHCP:** OFF (Static IP Address)
- **Static IP Address:** 10.0.0.2
- **Subnet Mask:** 255.255.255.0
- **Default Gateway:** 10.0.0.1

If the IP address is unknown or has been changed, follow the steps below to retrieve the IP address using a USB memory device:

1. Insert a USB memory device into one of the Type-A USB ports of the module and wait approximately ten seconds for the module to write an ip.txt file. This file contains the network configuration.
2. Plug in the USB memory device into your PC, select the USB drive, and open ip.txt file to retrieve the IP address.

### Dynamic IP Configuration

After the PC is properly configured with static IP and a good connection has been established, SCPI commands are used to configure the Ethernet settings to different values.

| Caution | Consult with your network administrator when configuring the network interface to avoid potential loss of access or discovery of the device. |
|---|---|

The following steps describe how to verify and set the Ethernet configuration through SCPI programming:

1. Using any SCPI client, send the `*IDN?` query command to your instrument IP address (10.0.0.2 default) on port 9001. Verify the response includes the model and serial numbers, and other information.

| Note | If a response is not returned, try switching to a cross-over Ethernet cable and send the query command again. Ensure that your SCPI client has the termination character enabled: \n Line Feed (xA). |
|---|---|

2. To configure the instrument for DHCP, send the following SCPI commands to your instrument IP address on port 9001:

   `SYSTem:COMMunicate:LAN:DHCP?`

   Queries the DHCP setting. The return value should be OFF if the instrument is currently configured for static IP.

   `SYSTem:COMMunicate:LAN:DHCP ON`

   Turns the DHCP setting ON. The setting is applied immediately upon sending the command.

3. Turning DHCP ON will change the IP address and the module must then be connected to a DHCP network to obtain a new IP address.
4. Obtain the new IP by one of the following:

   Use the Anritsu Discovery program to retrieve the new IP address (see "UDP Discovery Tool" on page 1-5 for instructions on using the discovery program).

   Insert a USB memory device into one of the USB Type-A ports to retrieve the ip.txt file containing the IP address information.
5. Send the *IDN? SCPI command to verify a connection has been established.

### UDP Discovery Tool

The UDP Discovery Tool is a Windows application that will display a sortable list of available MS2720xA devices on the network. This tool provides the ability to find and filter devices by Hostname, Model number, Serial number, MAC address, Version number, or by IP Address. If needed, the tool also provides the ability to save the results of your search and to view a brief history of your searches for comparison.

https://www.anritsu.com/en-US/test-measurement/support

During the download process, a screen will appear as shown in Figure 1-1.

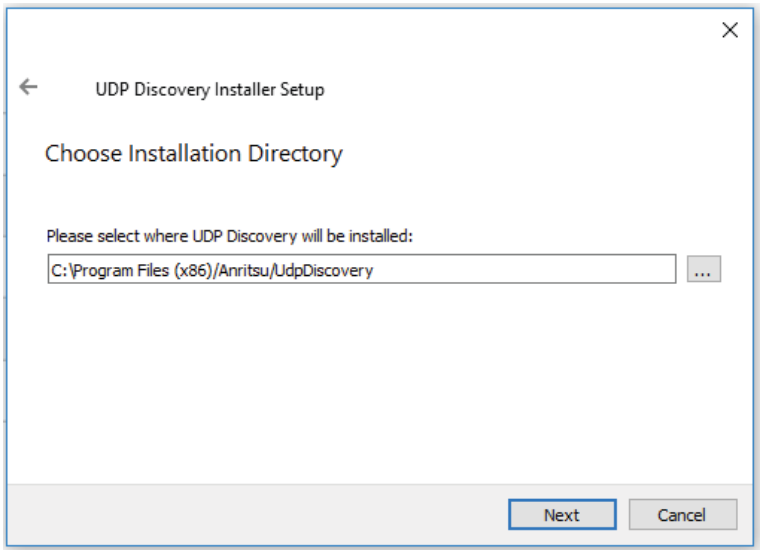
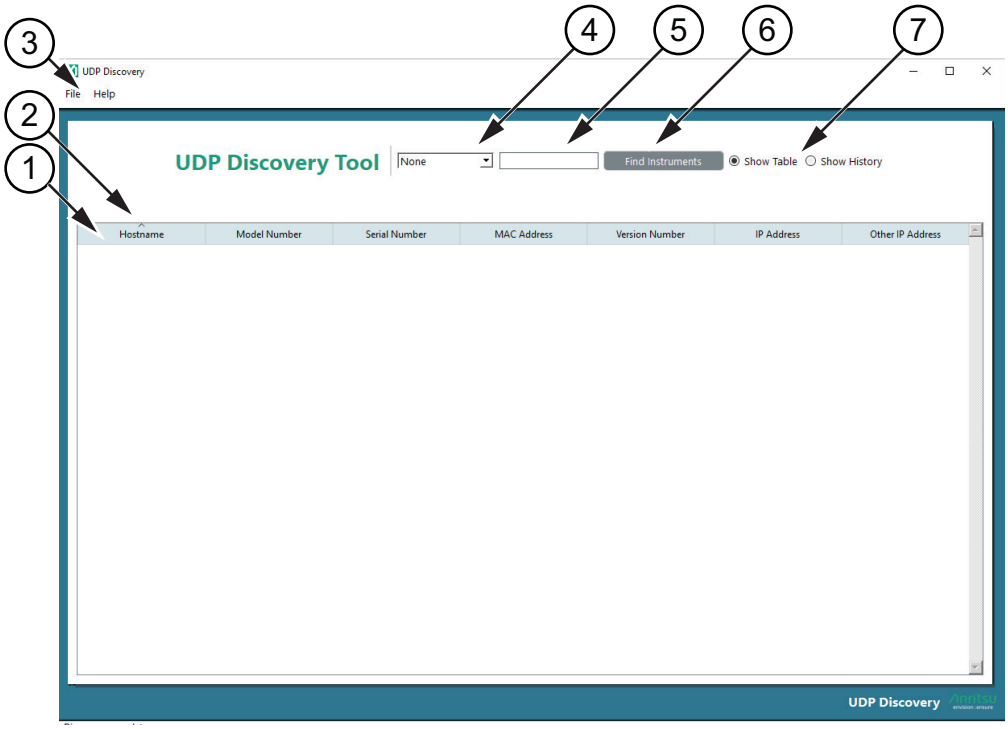



**Figure 1-1.** Installation Directory Default Location

When the download is complete, a user interface will appear as shown in Figure 1-2.




1. Column Header
2. Column Filter Caret
3. Toolbar
4. Selection Filter List
5. Filter Entry Field
6. Find Instruments Bar
7. Display Type

**Figure 1-2.** UDP Discovery Tool User Interface

### Communicating with the module

There could be many instruments already connected to the network. Use the selection filter to narrow the search results.

### Select a filter method

1. Click the connection arrow to display the filter list box are shown in Figure 1-3.

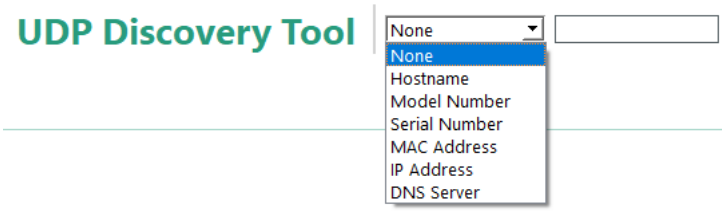



**Figure 1-3.** Filter List Box

2. Click a filter method from the list. The selections available are None, Hostname, Model Number, Serial Number, Mac Address, IP Address, IP Address, and DNS Server.

If a Hostname, Model Number, Serial Number, Mac Address, IP Address, IP Address, or DNS Server is not known, select None.

- None: All instruments attached to the local area network will be listed.
- Hostname: Host name of the probe
- Model Number: Model Number of the probe
- Serial Number: Serial Number of the probe
- MAC Address: Medium Access Control (MAC) address
- IP Address: Internet Protocol (IP) address
- DNS Server: Domain Name System (DNS) server

3. Click None to display an alphabetical listing of the inventoried instruments available to connect as shown in Figure 1-4.

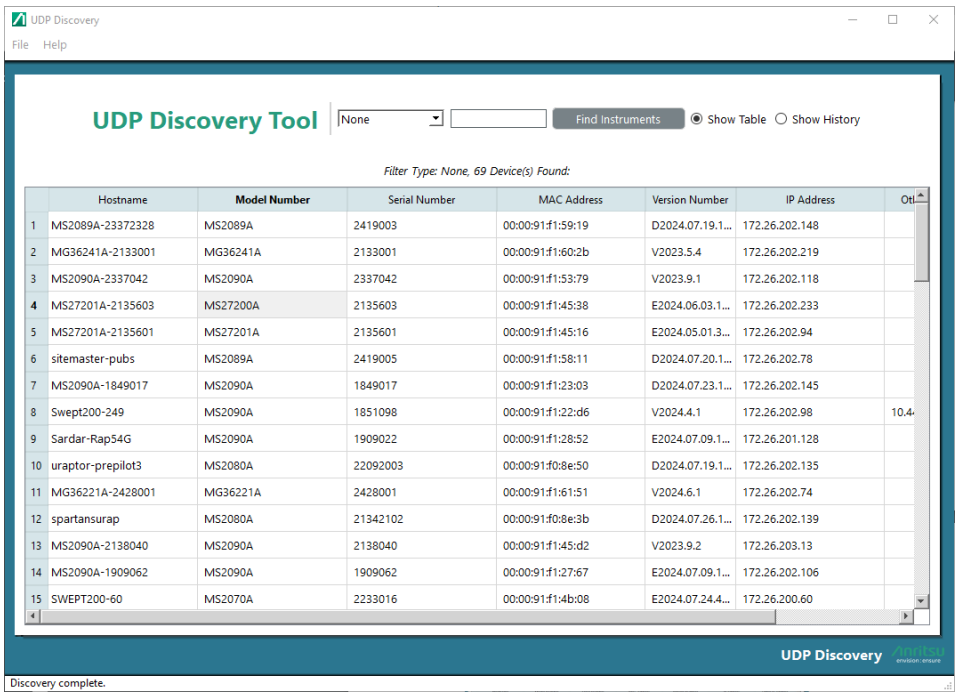



**Figure 1-4.** Displayed Instrument Inventory

### Sorting the Listing

Selecting a column header displays a filter caret above column header as shown in Figure 1-2. When the caret is clicked, the listing is sorted by the column in the direction the caret is pointing. Clicking the caret again reverses the caret direction and sorts the listing in reverse order.

### Copy and Paste Example

In this example, refer to the listing in Figure 1-4.

1. Click serial number 2135603.
2. Copy the serial number.
3. Paste the serial number in the box to the right of the filter list.
4. Click the Find Instruments bar. If the instrument is located, the instrument details will be displayed as shown in Figure 1-5.

A status readout is displayed above the column headings. In this example, the readout is:

*Filter Type: Serial Number, Filter Value: 1713003, 1 Device(s) Found:*

If the device could not be found, the last section would read: *No Device Found.*

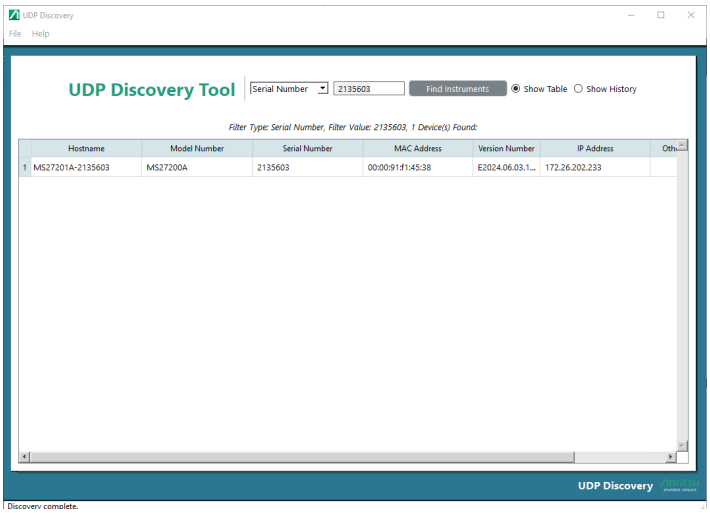



**Figure 1-5.** Locate Instrument

Connecting another device will erase the Show Table log and display the new device details. The previous erased device details will be available in the Show History log until the history details are erased. See"Clearing the History" on page 1-9.

### User Interface Display Types

The selectable radio buttons provide the Show Table and Show History user interface connection display.

**Show Table :** Click this radio button to display the device connection details as shown in Figure 1-5.

**Show History :** Click this radio button to display current and previous connection-activity details as shown in Figure 1-6.

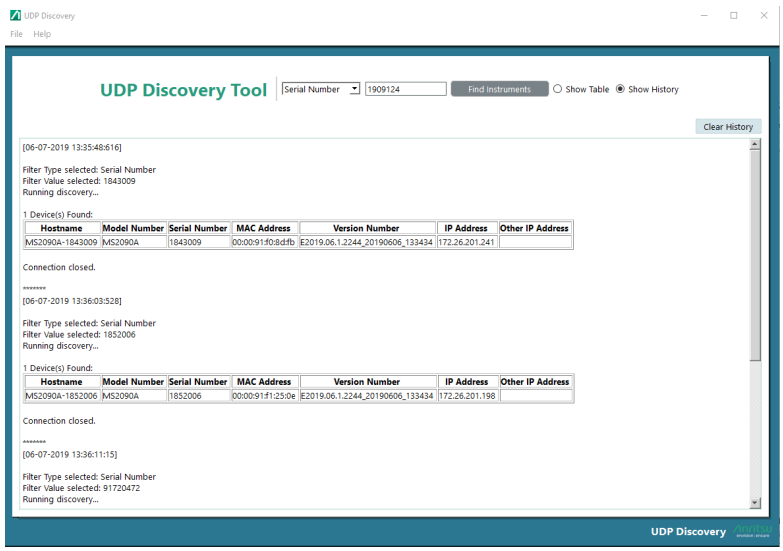



**Figure 1-6.** Show History

### Clearing the History

Clear the Show History log by clicking the Clear History button or by clicking Clear Log from the Toolbar File menu.

1. Click Clear History will display the dialog shown in Figure 1-7.

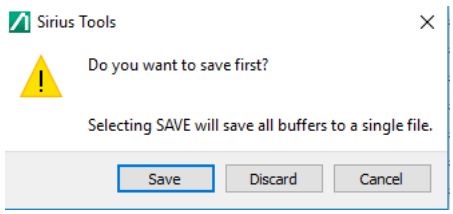



**Figure 1-7.** Clear History Dialog

   a. Click Save to save to a users selected location.
   b. Click Discard to remove the history contents without saving. The data will not be retrievable.
   c. Click Cancel to ignore the clear history request.

**2.** Click the File menu to display the menu as shown in Figure 1-8.

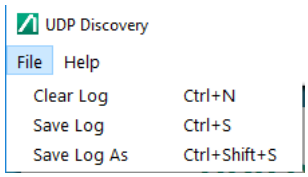



**Figure 1-8.** Clear History Using the File Menu

**a.** Click Clear Log to remove the History contents without saving. The data will not be retrievable.

**b.** Click Save Log to save to a users selected location. The location will be the same location selected from the Save Log As submenu.

**c.** Click Save Log As to save to a users selected location.

### Dynamic Configuration - Anritsu Remote Tool

Follow the instructions below to configure Ethernet settings (DHCP) on Anritsu Remote Tool for MS27201A:

1. Click System menu (three-line icon in upper left corner).
2. Click Settings to access the instrument settings menu, then select Ethernet to view the current network settings (IP address, HOSTNAME, etc.).

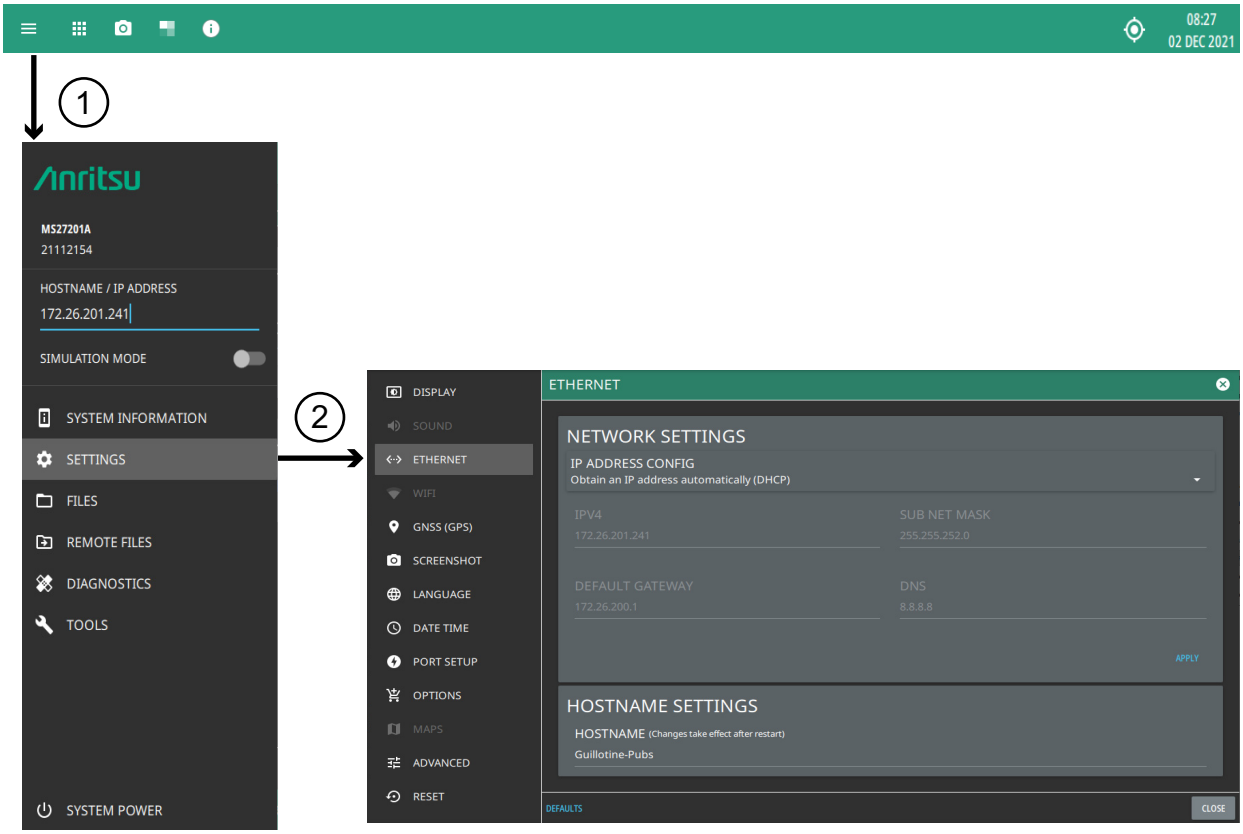



**Figure 1-9.** Ethernet Settings

The instrument IP address can be set automatically using DHCP or manually by entering the desired IP address, gateway address, and subnet mask. DHCP is an Internet protocol that automates the process of setting IP addresses for devices that use TCP/IP, and is the most common method of configuring a device for network use.

| | |
|---|---|
| **Note** | An active Ethernet cable must be connected to the instrument/module while it is turned on, a reboot may be required to establish a DHCP connection. If the port becomes inactive, verify that an active Ethernet cable is attached to the instrument, then cycle the instrument power off and on. |

## USBTMC Interface Connection and Setup (MS27200A)

The Universal Serial Bus (USB) architecture is a high-performance networking standard that is considered "plug and play" compatible. The USB driver software is automatically detected and configured by the operating system of the devices that are connected to the bus. The USB Type-C port is supported by MS27200A only.

The Type-C port can be used for remote program interface. This port conforms to the USB 3.0 standard and supports data rates of up to 5 Gbit/s with the following restrictions:

- One USB network can support up to 127 devices.
- The maximum length of USB cables between active devices is 3 meters for USB 3.0, 5 meters for USB 2.0, and 3 meters for USB 1.0.

The following USBTMC features are not yet supported:

- SET_FEATURE
- SYNCH_FRAME

For remote USB control, the controller PC must have a version of Virtual Instrument Software Architecture (VISA) installed that supports USBTMC (USB Test and Measurement Class) devices. Refer to http://www.ni.com/visa/ for software downloads and installation instructions.

The USB C port must be used to connect the MS27200A directly to a PC. The first time the module is connected to a PC, the normal USB device detection by the computer operating system takes place.

1. Power on the module and controller PC and wait for the systems to power up completely.
2. Connect the USB port C cable connector to the module.
3. Connect the USB port A cable connector to the controller PC USB host port. The controller PC generally installs the driver software in the background and briefly displays a notification balloon near the system tray when the device is ready for use and the setup is then complete.

   Some configurations will display a "New Hardware Found" dialog and you will then need to complete the remaining steps in this procedure by clicking Next.


**Figure 1-10.** Found New Hardware Wizard

4. Select to allow the Wizard to search for and install the USB software automatically.

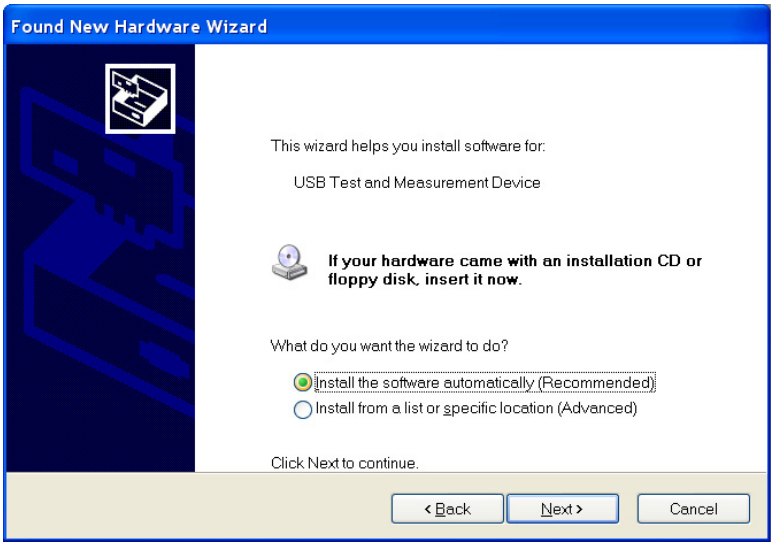



**Figure 1-11.** Found New Hardware Wizard

5. After the software finishes installing, close the wizard by clicking Finish.

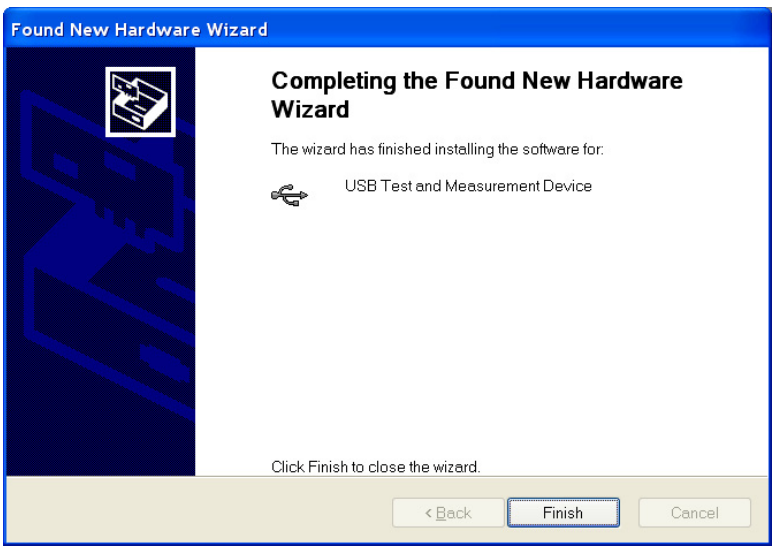



**Figure 1-12.** Found New Hardware Wizard

| **Note** | The USBTMC connection interface is not supported by instruments installed with Secure Communication, Option 17. MS27201A RSM is not equipped with USB Type-C port. |
|---|---|

# 1-4 Sending SCPI Commands with NI-VISA

SCPI commands can be sent through any VISA controller. VISA is a commonly used API in the Test and Measurement industry for communicating with units from a PC. The physical connection between the PC and the unit can be Ethernet or USB. NI-VISA is the National Instruments implementation of the VISA I/O standard. Information and downloads are available at http://www.ni.com/visa/.

The following example describes the verification that a VISA controller can interact with the unit. The images shown and the instructions and software may differ from the example shown here, so refer to the NI user instructions for more details. Note that the screenshots below are included to be shown as an example only.

Follow the instructions below to send SCPI commands via USBTMC standard or Ethernet:

1. On the PC, open VISA Interactive Control dialog and double-click on the desired instrument resource and press Enter for USBTMC.
2. Enter the IP address and press Enter for Ethernet

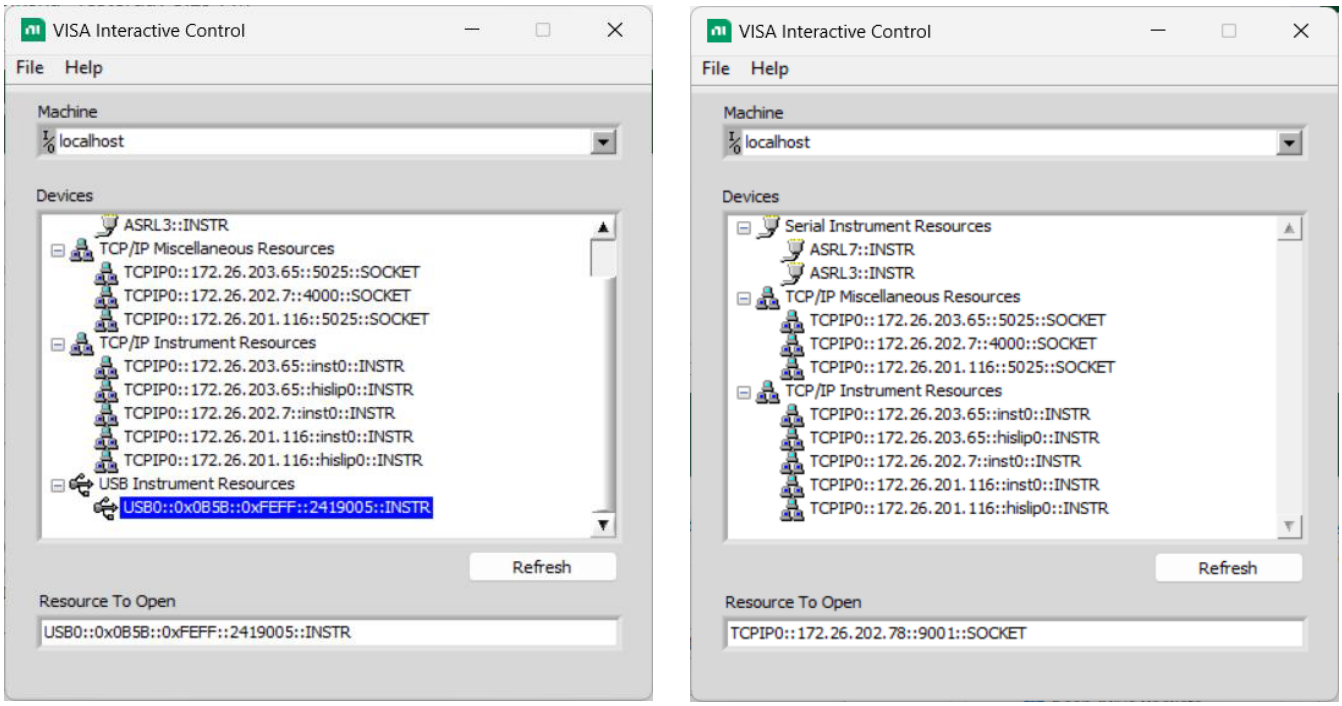



**Figure 1-13.** VISA Interactive Control Dialog - USBTMC (left); Ethernet (right)

3. Select Input/Output tab and enter the default `*IDN?\n command` and click Query button. Note that this step holds good for USBTMC and Ethernet.

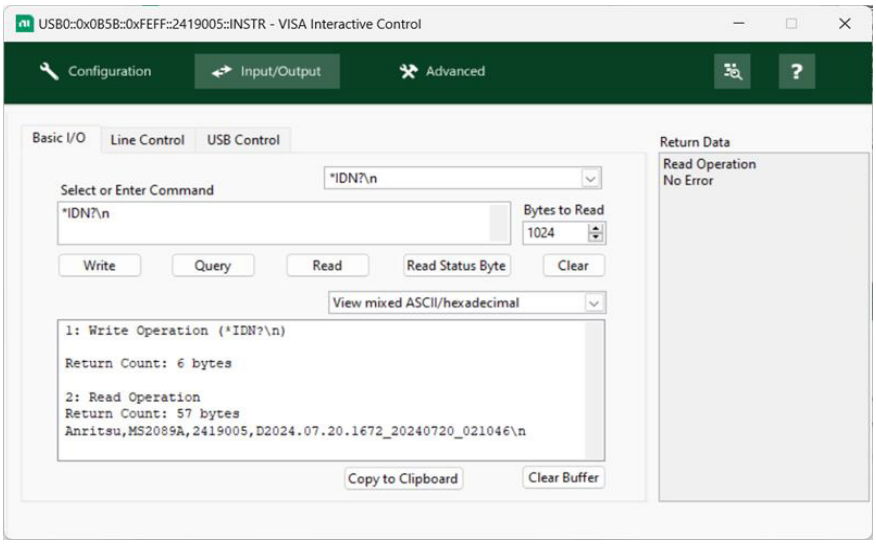



**Figure 1-14.** VISA Interactive Control - Input/Output Tab (USBTMC)

| **Note** | The NI-VISA controller is not compatible with instruments installed with Secure Communication Option 17. |
|---|---|

# 1-5 SCPI Command Overview

SCPI commands, which are also referred to as SCPI instructions, are messages to the instrument to perform specific tasks and to return data. The following sections provide descriptions of the command syntax and notational conventions, command types and the hierarchical command structure, and command subsystems and data parameter format. For further information about SCPI, refer to the Standard Commands for Programmable Instruments (SCPI) 1999.0 document hosted by the IVI Foundation. A subset of the standard conventions defined in that document as they apply to the MS2720xA are covered in this manual.

## Command Names

Typical SCPI commands consist of one or more keywords, parameters, and punctuation. SCPI command keywords can be a mixture of upper and lower case characters. As with any programming language, the exact command keywords and command syntax must be used. The syntax of the individual commands is described in detail in the programming command chapters. Unrecognized versions of long form or short form commands or improper syntax will generate an error.

### Long Format vs. Short Format

Except for common commands, each keyword has a long and a short form. The short form is a subset of the long form of the command. In this manual, the long form is presented with the short form in upper case and the remainder in lower case. For example, the long form of the command keyword to control the instrument display is `:DISPlay`.

The short form keyword is usually the first four characters of the long form (`:DISP` for `:DISPlay`). The exception to this rule is when the long form is longer than four characters *and* the fourth character is a vowel. In such cases, the vowel is dropped and the short form becomes the first three characters of the keyword. For example, the short form of the keyword `:POWer` is `:POW`.

A mixture of short form keywords with the entire long form keyword of each command is acceptable. For example, `:SENS:FREQuency:STAR` is an acceptable form of the command. However, `:SENS:FREQuen:STAR` is not an acceptable form of the command because `:FREQuen` is not the entire short nor long form of the command keyword.

## Command Syntax

The command statement usually starts with a colon ":" and includes a number of keywords and parameters. The leading colon is optional, but each subsequent keyword must be separated by a colon ":" and may be followed by a parameter setting when they are available. The end of the command string and the first command parameter must be separated by a space. For example:

```
:DISPlay:POINtcount 201
```

A comma "," is generally used for separating multiple command parameters, for example:

```
:SYSTem:DATE 2018,10,31
```

Complete command strings may be combined on a single line and separated by a semicolon ";" as follows:

```
:SENSe:FREQuency:STARt 1000; SENSe:FREQuency:STOP 5000
```

Some command keywords may have a keyword parameter suffix to differentiate between multiple features such as multiple trace options. When these features are available, the keyword parameter is identified and enclosed by curly braces. For example, `:TRACe<n>` is entered as `:TRACe1` or `:TRACe3`.

| | |
|---|---|
| **Note** | If a keyword parameter is not included in a command, the first option is implied. In the previous paragraph, `:TRACe` is identical to `:TRACe1`.<br>Angle brackets < > enclose keyword and command parameters. Curly braces { } enclose optional command parameters. Square brackets [ ] enclose optional command keywords. |

### Notational Conventions

The SCPI interface standardizes command syntax and style that simplifies the task of programming across a wide range of instrumentation. As with any programming language, the exact command syntax, and command keywords, must be used. Unrecognized commands or improper syntax will not function and most often produce an error condition. The table below lists the standard notational conventions used in this document.

**Table 1-2.** Notational Conventions

| | |
|---|---|
| : | A colon links command keywords together to form commands. The colon is not an actual part of the keyword, but is a signal to the SCPI interface parser. A colon must precede a root keyword immediately following a semicolon (see "Notational Examples" on page 1-26). |
| ; | A semicolon separates commands if multiple commands are placed on a single program line. |
| [ ] | Square brackets enclose one or more optional keywords. |
| { } | Curly braces enclose one or more keyword or command parameters that may be included one or more times. |
| \| | A vertical bar indicates "or" and is used to separate alternative parameter options.<br>For example: `ON\|OFF` is the same as `ON` or `OFF`. |
| < > | Angle brackets enclose keyword and command parameters (see "Data Parameters" on page 1-18). |

### Command Terminators

The new line character (ASCII 10) in the last data byte of a command string is used as a command terminator. Use of a command terminator will reset the command path to the root of the tree.

## Data Parameters

Data parameters, referred to simply as "parameters," are the quantitative values used as arguments for the SCPI commands. The parameter type associated with a particular SCPI command is determined by the type of information required to control the particular function. For example, Boolean (`ON|OFF`) type parameters are used with commands that control switch functions. Some command descriptions specify the type of data parameter to be used with each command. The most commonly used parameter types are numeric, extended numeric, discrete, string, and Boolean.

Numeric: Numeric parameters comprise integer numbers or any number in decimal or scientific notation, and may include polarity signs. This includes specific extended values as defined in "Data Parameter Notations" below.

Extended Numeric: Extended numeric parameters include values such as `NAN|INFinity|NINFinity` (not a number|infinity|negative infinity).

Discrete: Discrete parameters, such as `INTernal` and `EXTernal`, are used to control program settings to a predetermined finite value or condition.

Boolean: Boolean parameters represent binary conditions and may be expressed as `ON`, `OFF` or `1`, `0`.

### Data Parameter Notations

The following conventions are used to note data parameter descriptions:

**Table 1-3.** Parameter Notations

| | |
|---|---|
| <integer> | an unsigned integer without a decimal point (implied radix point) |
| <numeric_value> | SCPI numeric value: `<NRf>\|NAN\|INFinity\|NINFinity` or other types<br><br><NRf> is any of <NR1>\|<NR2>\|<NR3><br><NR1> is a signed integer without a decimal point (implied radix point)<br><NR2> is a signed number with an explicit radix point<br><NR3> is a scaled explicit decimal point numeric with an exponent (floating point number)<br><br>+/- infinity (`INFinity\|NINFinity`) and Not-A-Number (`NAN`) values can be specified for parameters of type NR3 with the values of +/- 9.9e37 and 9.91e37, respectively. |
| <char> | <CHARACTER PROGRAM DATA> Examples: CW, FIXed, UP, and DOWN |
| <string> | <STRING PROGRAM DATA> ASCII characters enclosed by double quotes, such as `"OFF"` |
| <block> | IEEE-488.2 block data format (see "Spectrum Trace Data Format") |

## Spectrum Trace Data Format

Trace data uses SCPI standard (IEEE 488.2) block data format. The data format is '#AXD', where D is a comma separated list of amplitudes (in ASCII), X is one or more ASCII digits specifying the number of bytes in D, and A is a single ASCII digit specifying the number of digits in X.

Trace data only contains amplitude. The frequency information for each point is

```
Frequency = start_frequency + (span/(display_points-1))*N
where N = 0, 1, 2, … display_points
```

## Unit Suffixes

Unit suffixes are not required for data parameters, provided the values are scaled for the global default units; they do not require a space and they are not case sensitive. The SCPI default units are:

- `HZ` (hertz) for frequency related parameters
- `DB` for power related parameters
- `MS` (milliseconds) for time related parameters
- `M` (meters) for distance related parameters

## Query Commands

All commands, unless specifically noted in the commands syntax descriptions, have a query form. As defined in IEEE-488.2, a query is a command with a question mark "?" appended to the command. For example:

```
SENS:FREQ:STAR?
```

When a query form of a command is received, the current setting associated with the command is placed in the output buffer. Query commands always return the short form of the parameter unless otherwise specified. Boolean values are returned as `1` or `0`, even when they can be set as `on` or `off`.

## Hierarchical Command Structure

All SCPI commands, except the common commands, are organized in a hierarchical structure similar to the inverted tree file structure used in most computers. The SCPI standard refers to this structure as "the Command Tree." The command keywords that correspond to the major control functions are located at the top of the command tree. The root command keywords for the SCPI command set are shown in Figure 1-15.

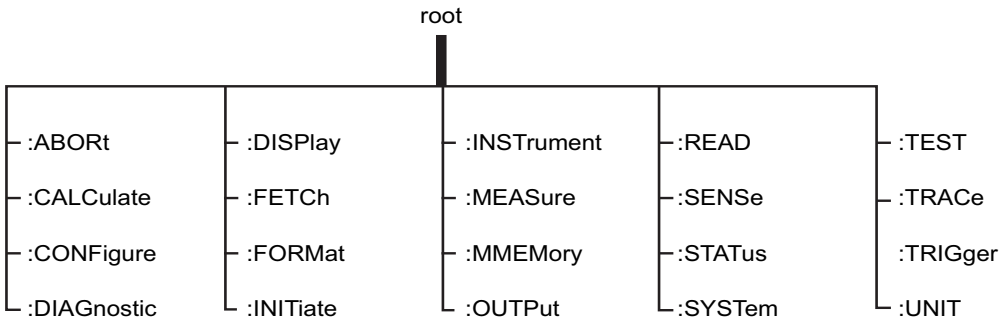



**Figure 1-15.** SCPI Command Tree

All SCPI commands, except the `:ABORt` command, have one or more subcommands (keywords) associated with them to further define the function to be controlled. The subcommand keywords may also have one or more associated subcommands (keywords). Each subcommand level adds another layer to the command tree. The command keyword and its associated subcommand keywords form a portion of the command tree called a command subsystem. A sample of the `:UNIT` command subsystem is shown in Figure 1-16.

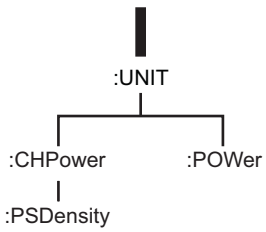



**Figure 1-16.** SCPI :Sample :UNIT Subsystem

# 1-6 SCPI Command Types

The MS2720xA command set includes the following command types:

- SCPI “common” commands (IEE488.2 mandated commands)
- SCPI required commands (per SCPI 1999.0)
- SCPI optional commands

## SCPI Common Commands

Some common commands are defined in the IEEE-488.2 standard and must be implemented by all SCPI compatible instruments/modules. These commands are identified by an asterisk (*) at the beginning of the command. These commands are defined to control status registers, status reporting, synchronization, and other common functions. Examples of the supported common commands are shown below.

***CLS**

Title: Clear Status Command

Description: This command clears all status data structures in the device (registers and error queue) and forces the Operation Complete state to Idle.

***IDN?**

Title: Identify

Description: This command returns the following information in <string> format separated by commas: manufacturer name ("Anritsu"), model number, serial number, and firmware package number.

***RST**

Title: Reset

Description: This command sets the unit into a known state that is ready for SCPI command processing. Setting into a known state includes the following steps:

1. All applications beside the default application of the particular model will be turned off.

2. Settings for all applications (regardless of an application being active) are preset to default values.

3. System settings excluding settings associated with client connections (like ethernet addresses or DNS) are preset to default values.

4. Pending operations are aborted and the pending operation flag will be reset. The unit will be forced into the operation complete command idle state (OCIS) and the operation complete query idle state (OQIS) before the pending operation flag is reset.

### SCPI Required Commands

The required SCPI commands are listed below. These commands work in all measurement modes.

**Table 1-4.** SCPI Required Commands

| |
|---|
| :STATus |
| :SYSTem |

### SCPI Optional Commands

Optional SCPI commands comprise the majority of the MS2720xA command set. These commands control most of the programmable functions. The top level command subsystems are listed in the table below.

**Table 1-5.** SCPI Optional Commands

| | | | |
|---|---|---|---|
| :ABORt | :FETCh | :MMEMory | :SWEep |
| :CALCulate | :FORMat | :OUTPut | :TEST |
| :CONFigure | :INITiate | :READ | :TRACe |
| :CPRI | :INPut | :ROUTE | :TRIGger |
| :DIAGnostic | :INSTrument | :SENSe | :UNIT |
| :DISPlay | :MEASure | | |

The SCPI optional commands are sorted by measurement modes and the commands may be repeated in more than one mode.

## 1-7 Status System Reporting

The MS2720xA status system consists of the following SCPI-defined status-reporting structures:

- The Instrument Summary Status Byte Group
- The Standard Event Status Group
- The Operation Status Group
- The Questionable Status Group

The following paragraphs describe the registers that make up a status group and explain the status information that each status group provides.

> **Note** Parallel Polling is not supported in the MS2720xA.

### Status Group Registers

In general, a status group consists of a condition register, a transition filter, an event register, and an enable register. Each component is briefly described in the following paragraphs.

#### Condition Register

The condition register is continuously updated to reflect the current status of the MS2720xA. There is no latching or buffering for this register, it is updated in real time. Reading the contents of a condition register does not change its contents.

### Event Register

The event register latches transition events from the condition register as specified by the transition filter. Bits in the event register are latched, and once set they remain set until cleared by a query or a *CLS command Event registers are read only.

### Enable Register

The enable register specifies the bits in the event register that can produce a summary bit. The MS2720xA logically ANDs corresponding bits in the event and enable registers, and ORs all the resulting bits to obtain a summary bit. Summary bits are recorded in the Summary Status Byte. Enable registers are read-write. Querying an enable register does not affect it.

## Status Group Reporting

The state of certain MS2720xA hardware and operational events and conditions can be determined by programming the status system. Three lower status groups provide status information to the Summary Status Byte group. The Summary Status Byte group is used to determine the general nature of an event or condition and the other status groups are used to determine the specific nature of the event or condition. The following paragraphs explain the information that is provided by each status group. Programming commands for the status system can be found in Section 2-1 “Common Commands”.

### Summary Status Byte Group

The Summary Status Byte group, consisting of the Summary Status Byte Enable register and the Summary Status Byte, is used to determine the general nature of a MS2720xA event or condition. The bits in the Summary Status Byte provide the following:

**Table 1-6.** Status Byte Group

| Bit # | Bit Name | Description |
|---|---|---|
| **0,1** | Not Used | These bits are always set to 0. |
| **2** | Not Used | These bits are always set to 0. |
| **3** | Not Used | These bits are always set to 0. |
| **4** | Message Available (MAV) | Set to indicate that the MS2720xA has data ready in its error queue. |
| **5** | Standard Event (ESB) | Set to indicate that the Standard Event Status summary bit has been set. The Standard Event Status register can then be read to determine the specific event that caused the bit to be set. |
| **6** | Master Summary Status (MSS/RQS) | Set to indicate that the MS2720xA has at least one reason to require service. This bit is also called the Master Summary Status Bit (MSS). The individual bits in the Status Byte are ANDed with their corresponding Service Request Enable Register bits, then each bit value is ORed and input to this bit. |
| **7** | Not Used | These bits are always set to 0. |

### Standard Event Status Group

The Standard Event Status group, consisting of the Standard Event Status register (an Event register) and the Standard Event Status Enable register, is used to determine the specific event that set bit 5 of the Summary Status Byte. The bits in the Standard Event Status register provide the following:

**Table 1-7.** Standard Event Status Group

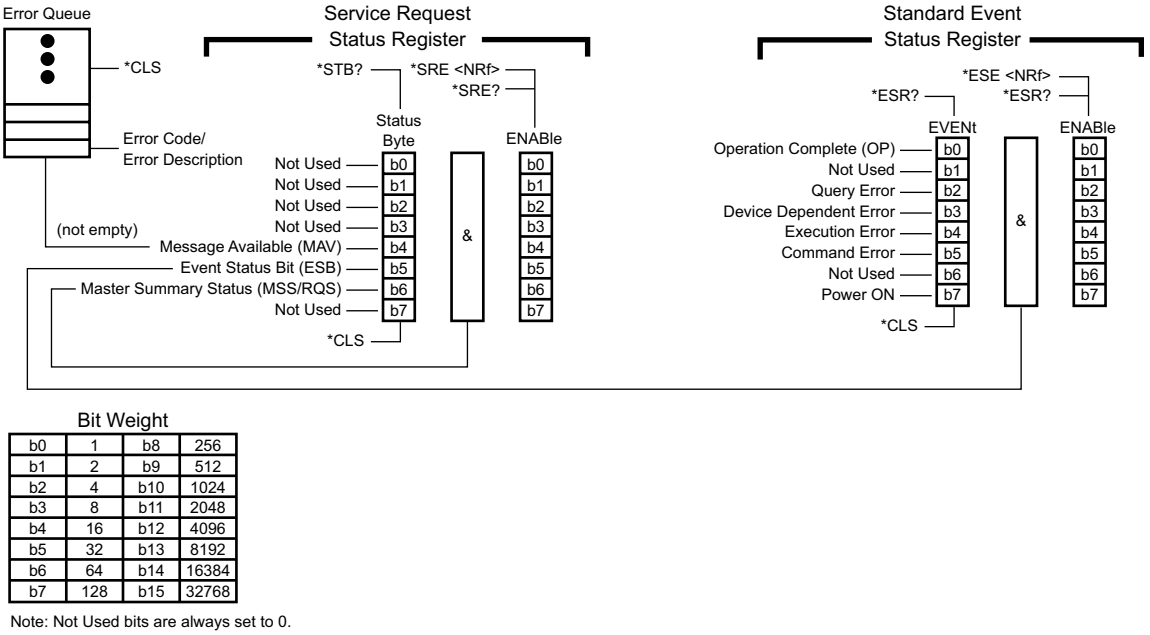
| Bit # | Bit Name | Description |
|---|---|---|
| 0 | Operation Complete (OP) | Set to indicate that all pending MS2720xA operations were completed following execution of the "*OPC" command.<br>For more information, see the descriptions of the *OPC, *OPC?, and *WAI commands in Section 2-1 "Common Commands". |
| 1 | Not Used | The bit is always set to 0. |
| 2 | Query Error | Set to indicate that a query error has occurred (error -400 in Appendix A, "SCPI Error Table"). |
| 3 | Device Dependent Error | Set to indicate that a device-dependent error has occurred (errors -300 to -399 in Appendix A, "SCPI Error Table"). |
| 4 | Execution Error | Set to indicate that an execution error has occurred (errors -200 to -299 in Appendix A, "SCPI Error Table"). |
| 5 | Command Error | Set to indicate that a command error (usually a syntax error) has occurred (errors -100 to -199 in Appendix A, "SCPI Error Table"). |
| 6 | Not Used | This bit should be set to 0 (zero). |
| 7 | Power ON | Set to indicate that the MS2720xA is powered ON and in operation. |

### Operation Status Group

The Operation Status group, consisting of the Operation Condition register, the Operation Positive Transition register, the Operation Negative Transition register, the Operation Event register, and the Operation Event Enable register, is used to determine the specific condition that set bit 7 in the Summary Status Byte. The bits in the Operation Event register provide the following:

**Table 1-8.** Operation Status Group

| Bit # | Bit Name | Description |
|---|---|---|
| 0-7 | Not Used | This bit should be set to 0 (zero). |
| 8 | Sweep Complete | Set to indicate that a sweep is complete. |
| 9 | I/Q Capture | Set to indicate that I/Q data capture is in process. |
| 10-15 | Not Used | These bits should be set to 0 (zero). |




Bit Weight

| | | | |
|---|---|---|---|
| b0 | 1 | b8 | 256 |
| b1 | 2 | b9 | 512 |
| b2 | 4 | b10 | 1024 |
| b3 | 8 | b11 | 2048 |
| b4 | 16 | b12 | 4096 |
| b5 | 32 | b13 | 8192 |
| b6 | 64 | b14 | 16384 |
| b7 | 128 | b15 | 32768 |

Note: Not Used bits are always set to 0.

**Figure 1-17.** Status Register Structure

# 1-8 SCPI Commands

Command subsystems control specific functions and some general purpose functions as described earlier in this text. All command subsystems are identified by the leading keyword in a command statement, as in `:INITiate:CONTinuous`.

The following information is provided for each command described in this guide:

- The command name, see "Command Names" on page 1-16
- The path from the subsystem root command, see "Hierarchical Command Structure" on page 1-19
- The query form of the command (if applicable), see "Query Commands" on page 1-19
- A description of the command function and purpose
- The data parameters used as arguments for the command, see "Data Parameters" on page 1-18 This may include the parameter type, available parameter choices, and numeric ranges and default values.

Refer to the following for the programming command descriptions:

- Chapter 2, "MS2720xA Commands"

Refer to the following for an alphabetical list of all supported SCPI commands:

- Appendix A, "Alphabetical SCPI Command List"

Refer to the next section for the programming examples.

# 1-9 SCPI Command Programming Examples

SCPI commands are sent to port 9001 of the instrument/module. Command statements read from left to right and from top to bottom. In the command statement below, the `:FREQuency` keyword immediately follows the `:SENSe` keyword with no separating space. A space is used between the command string and its argument.

## Notational Examples

**`[:SENSe]:FREQuency:STARt <numeric_value>{HZ|KHZ|MHZ|GHZ}`**

The following command syntax all produce the same result:

```
:SENSe:FREQuency:STARt 1 MHZ
:SENS:FREQ:STAR 1 MHZ
:sense:frequency:start 1000000
:FREQ:STAR 1000 KHZ
```

| **Note** | Note that the first keyword in the command string does not require a leading colon; however, it is good practice to always use a leading colon for all keywords. Note also that the :SENSe keyword is optional. This is a SCPI convention for all voltage or signal source type instruments that allows shorter command statements to be used. |
|---|---|

**`:CALCulate:MARKer<n>:X <numeric_value>{HZ|KHZ|MHZ|GHZ}`**

The first two commands below set the location of marker 1, the third command sets the location of marker 2:

```
:CALC:MARK:X 1 GHZ
:CALC:MARK1:X 1 GHZ
:CALC:MARK2:X 2ghz
```

**`:UNIT:POWer <DBM|DBUV>`**

The following command syntax is identical:

```
:UNIT:POWer DBM
:unit:pow dbm
```

**`:INITiate:CONTinuous <0|1|ON|OFF>`**

The following commands are identical:

```
:INITiate:CONTinuous OFF
:init:cont 0
```

The following is an example of a multiple command statement that uses two separate commands in a single program line:

```
:FREQuency:STARt 10E6;:FREQuency:STOP 20E9
```

| **Note** | A semicolon is used to join the commands and a leading colon used immediately after the semicolon to start the second command. |
|---|---|

## Spectrum Trace Data via SCPI

This section provides an abbreviated example of setting up and capturing spectrum trace data via SCPI commands. SCPI commands are sent to port 9001.

```
//Set the Start and Stop Frequencies
SENS:FREQ:STAR 88 MHz
SENS:FREQ:STOP 108 MHz

//Set the RBW to 30 kHz
BAND:RES 30 KHz

//Set the Reference Level to -30 dBm
DISP:WIND:TRAC:Y:SCAL:RLEV -30

//Set to single sweep
INIT:CONT OFF

//Get trace amplitude data
TRACE:DATA? 1

//Get number of display points to calculate frequency array
DISP:POIN?
```

## C/C++

This example is run on the command line using the ethernet raw socket connection. It sends the *IDN? query to the instrument/module and prints the response to the console.

```
// IdnExample.cpp : Microsoft Visual Studio-Generated Example
//Usage : IdnExample "TCPIP0::172.26.202.100::9001::SOCKET"
//where xxxxxxxx_xxx_xx is the USB Device ID of the instrument.
//Output : The string identity string returned from the instrument.
//VISA Header : visa.h (must be included)
//VISA Library : visa32.lib(must be linked with)
#include "stdafx.h"
#include "stdio.h"
#include "string.h"
#include "visa.h"
#define BUFFER_SIZE 255
int main(int argc, char* argv[])
{
ViStatus status; /* For checking errors */
ViSession defaultRM, instr; /* Communication channels */
ViUInt32 retCount; /* Return count from string I/O */
ViChar buffer[BUFFER_SIZE]; /* Buffer for string I/O */
char tempDisplay[BUFFER_SIZE]; /* Display buffer for example */
char *pAddress;
/* Make sure we got our address. */
if ( argc < 2 )
{
printf("Usage: IdnExample
\"TCPIP0::172.26.202.100::9001::SOCKET");
printf("\t where xxxxxxxx_xxx_xx is the USB Device ID of your instrument.\n");
return -1;
}
/* Store the address. */
pAddress = argv[1];
/* Begin by initializing the system*/
status = viOpenDefaultRM(&defaultRM);
if (status < VI_SUCCESS)
{
/* Error Initializing VISA...exiting*/
printf("Can't initialize VISA\n");
return -1;
}
```

```
/* TCPIP0::172.26.202.100::9001::SOCKET*/
/* NOTE: For simplicity, we will not show error checking*/
/* TODO: Add error handling. */
status = viOpen(defaultRM, pAddress, VI_NULL, VI_NULL, &instr);
/* Set the timeout for message-based communication*/
/* TODO: Add error handling. */
status = viSetAttribute(instr, VI_ATTR_TMO_VALUE, 120000);
/* Ask the device for identification */
sprintf(buffer, "*IDN?\n");
status = viWrite(instr, (unsigned char *)&buffer[0], 6, &retCount);
status = viRead(instr, (unsigned char *)buffer, BUFFER_SIZE, &retCount);
/* TODO: Add code to process data. */
strncpy(tempDisplay, buffer, retCount);
tempDisplay[retCount] = 0; /* Null-terminate display string. */
printf("*IDN? Returned %d bytes: %s\n", retCount, tempDisplay);
/* Close down the system */
/* TODO: Add error handling. */
status = viClose(instr);
status = viClose(defaultRM);
return 0;
}
```

## IQ Capture Programming Examples

### Raw Socket Connection

import socket from time import sleep, time

```
class SocketConnection:

"""Provides a means to connect and send SCPI commands to the DUT using a raw TCP
socket."""

def __init__(self, ipAddress):
"""Initializes an instance of SocketConnection class
@param ipAddress The IP address of the device"""

# split out port number if given
splitIpAddress = ipAddress.split(':')

assert len(splitIpAddress) > 0
assert len(splitIpAddress) <= 2
self._ipAddress = splitIpAddress[0]

#assign port
if len(splitIpAddress) == 2:
self._portNumber = int(splitIpAddress[1])
else:
self._portNumber = 9001

self._socketConnection = None

self._timeoutInSec = 120
self._socketReadSize = 4096
self.__nonBulkDataSizeCuttoff = 32768
# Time to let the other end of the connection close
self.__timeoutAfterCloseInSec = 1
self._terminatedBlockResponse = False
self.prefix = ''
self._verbose = False

self._establishConnection()
```

```
def __del__(self):
"""This gets called by the garbage collector so it is possible that the connection
will remain open for a while before this gets collected."""
self._closeConnection()

def getpeername(self):
return self._ipAddress, self._portNumber

def settimeout(self, *args, **kwargs):
return self._socketConnection.settimeout(*args, **kwargs)

def expectTerminatedBlockResponse(self, newval=None):
if newval is not None:
self._terminatedBlockResponse = newval
return self._terminatedBlockResponse

def sendWriteCommand(self, scpiCommand):
"""Sends a SCPI write command. @param scpiCommand The SCPI command to send."""

scpiCommand = self.prefix + scpiCommand
try:
returnValue = self._socketConnection.sendall(scpiCommand + "\n")
assert returnValue is None, "Error sending command: " + scpiCommand
if self._verbose:
if len(scpiCommand) < self.__nonBulkDataSizeCuttoff:
print(scpiCommand + " sent successfully")
else:
print( "sent long scpi command of length: " + str(len(scpiCommand)))
except socket.error as  msg:
assert False, "Failed to send SCPI command: a socket error occurred (Error code: "
+ str(msg[0]) + ", Error message: " + str(msg[1]) + ")"
return
```

```
def sendQueryCommand(self, scpiCommand):
"""Sends a SCPI query command and return the response. @param scpiCommand The SCPI
query to send. @return The result of the SCPI command."""
scpiCommand = self.prefix + scpiCommand
try:
returnValue = self._socketConnection.sendall(scpiCommand + "\n")
assert returnValue is None, "failed to send command"
if self._verbose:
print(scpiCommand + " sent successfully")

# Read 1 byte to check for a block data response header
data = self._socketConnection.recv(1)
assert len(data) > 0, "No data returned for query"
if len(data) > 0 and data[0] == '#':
# Block data response
data = self._getBlockDataResponse()
elif len(data) > 0 and data[0] == '\n':
# Check for a response string that only contains a newline. Remove the newline and
return empty data.
data = data[:-1]
elif len(data) > 0:
# ASCII response: receive until the entire response is read
while True:
data += self._socketConnection.recv(self._socketReadSize)

assert len(data) < self.__nonBulkDataSizeCuttoff, \
"No newline character found in response to "  + scpiCommand + " SCPI command."

# Check for a new line at the end of the response
if data[-1] == '\n':
break;

# Remove the trailing \n from the response
data = data[:-1]
if self._verbose:
print('Data received: "%s"' % data)
```

```
except socket.error as msg:
assert False, "Failed to send SCPI command: a socket error occurred \n" +
msg.__str__()
return data

def _establishConnection(self):
"""Establishes a connection. The call will fail if a connection is already open."""
assert self._socketConnection is None, "connection should not already be open"
try:
self._socketConnection = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
self._socketConnection.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
self._socketConnection.settimeout(self._timeoutInSec)
self._socketConnection.connect((self._ipAddress, self._portNumber))
self._socketConnection.setsockopt(socket.IPPROTO_TCP, socket.TCP_NODELAY, 1)
except socket.error as msg:
assert False, "Failed to establish DUT connection (Error code: " + str(msg[0]) + ",
Error message: " + str(msg[1]) + ")"

def _closeConnection(self):
"""Closes the socket connection and asserts that it closed. This informs the other
end of the socket that it should close but it may take some time depending on
implementation, network conditions, etc."""
if self._socketConnection is not None:
self._socketConnection.shutdown(socket.SHUT_RDWR)
self._socketConnection.close()
self._socketConnection = None
sleep(self.__timeoutAfterCloseInSec)
assert self._socketConnection is None, "Socket connection not closed"

def _getBlockDataResponse(self):
""" Receives a SCPI block data response of the form 'AXD' where A is a single ASCII
byte specifying the number of digits in X, X is one or more ASCII bytes specifying
the number of bytes in D, and D is one or more bytes containing the response binary
data."""
numSizeBytes = int(self._socketConnection.recv(1))

assert numSizeBytes > 0, "The definite-length empty block response must be #10 not
#0."
```

```
numDataBytesLeft = int(self._socketConnection.recv(numSizeBytes))
responses = []
readBuffer = bytearray(numDataBytesLeft)
view = memoryview(readBuffer)

timeoutSeconds = self._socketConnection.gettimeout()
lastReadTime = time()

while numDataBytesLeft > 0:
numBytesRead = self._socketConnection.recv_into(view, numDataBytesLeft)
if numBytesRead > 0:
lastReadTime = time()

dt = time() - lastReadTime
if dt > timeoutSeconds:
raise Exception('Timeout after %d ms: Only read %d/%d bytes'
% (dt, len(readBuffer),
len(readBuffer) + numDataBytesLeft))

view = view[numBytesRead:]
numDataBytesLeft = numDataBytesLeft - numBytesRead

if self._terminatedBlockResponse:
blockTerminator = self._socketConnection.recv(2)
assert blockTerminator in ('\r\n', '\n')

if self._verbose:
print("Read bytes of block data: ", len(readBuffer))
return readBuffer

def reset(self, delay_seconds=-1):
"""Resets the established connection @param delay_seconds: Wait time between
closing the connection and attempting tore-establish the connection.  This is
useful when rebooting an instrument."""
self._closeConnection()
```

```
if delay_seconds >= 0:
sleep(delay_seconds)
try:
self._establishConnection()
except socket.error as msg:
assert False, "Failed to establish DUT connection (Error code: " + str(msg[0]) + ",
Error message: " + str(msg[1]) + ")"
else:
reset_timeout = 300  # 300 seconds == 5 minutes == max polling time
time.sleep(5)  # Fixed delay before attempting to reconnect
while reset_timeout > 0:
try:
self._socketConnection = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
self._socketConnection.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
self._socketConnection.settimeout(self._timeoutInSec)
self._socketConnection.connect((self._ipAddress, self._portNumber))
break
except Exception as msg :
self._socketConnection.close()
self._socketConnection = None
sleep(1)
reset_timeout -= 1
if reset_timeout <= 0:
assert False, "Failed to establish DUT connection (Error code: " + str(msg[0]) + ",
Error message: " + str(msg[1]) + ")"
```

### IQ Block Capture via SCPI

```
SENS:FREQ:CENTER 100 MHz
SENS:FREQ:SPAN 20 MHz
SWEEP:MODE FFT
//Set RBW 30 kHz
BANDWIDTH 30 KHz
//Set Reference Level to -30 dBm
DISP:WIND:TRAC:Y:SCAL:RLEV -30
//Set to single sweep
INIT:CONT OFF
//abort any sweep in progress
:ABORT


//Set Capture bandwidth. Not same as RBW.
[:SENSe]:IQ:SAMPle SB2


//Set 16 bit resolution
IQ:BITS I16


//Set to IQ block capture mode
IQ:MODE SINGLE
//enable time stamp
SENS:IQ:TIME 1


//Set capture length to 5 msec
IQ:LENGTH 5 ms


//Start IQ Capture. Triggers single capture. Data is saved to DDR2 SDRAM memory.
MEAS:IQ:CAPT


//Check if capture is completed normally
STATus:OPERation?


//The STATus:OPERation? query responds with a integer. Convert this integer to
binary.
//Bit 9 is set to 1 when the MEAS:IQ:CAPT command is issued.
//Bit 9 is set to 0 when the capture is completed normally in block mode.
```

### IQ Capture Data to Absolute Power Level

This is a sample Matlab/Octave program that shows how Raw IQ capture data can be related to an Absolute power level.

```
%Copy data into captureData array
%Separate the data and build the complex IQ vector.
%First column contains Q and the second I
quadphase   = captureData(:,1);
inphase = captureData(:,2);
IQData = (inphase+1i*quadphase);
%Send SCPI Command [:SENSe]:iQ:SAMPle:CALibration:CONFiguration?
%and get absolute reference offset
absolute_ref_offset = -2.007958;
fs = 122.88e6;%Sampling frequency
n = 1024; %number of samples
%Perform fft
y = abs(fft(IQData, n));
y = fftshift(y);
%Scale fft output
y = y/n;
%To power
y = 20 * log10(sqrt(10 * y);
%To Absolute power level
y = y + absolute_ref_offset;
%Peak Value
peak = max(y);
f = fs*(-n/2:n/2-1)/n;
plot(f, y);
xlabel("Frequency in Hz"); % x-axis label
ylabel("Power in dBm"); % y-axis label
```

# Chapter 2 — MS2720xA Commands

## 2-1 Common Commands

***CLS**

Title: Clear Status Command

Description: This command clears all status data structures in the device (registers and error queue) and forces the Operation Complete state to Idle.

***ESE <numeric_value>**
***ESE?**

Title: Standard Event Status Enable

Description: This command provides access the Standard Event Status Enable Register.

Refer to the Status System Reporting section within this manual or IEEE 488.2, section 11.5.1 for more information about each supported bit.

Value ranges from 0 to 255.

Query returns a NR1 Numeric response.

Refer to *ESR? for the list of supported bit values.

Parameter(s): <numeric_value>

***ESR?**

Title: Standard Event Status

Description: This command queries the Standard Event Status Register.

Refer to the Status System Reporting section within this manual or IEEE 488.2, section 11.5.1 for more information about each supported bit.

Query returns a NR1 Numeric response.

Supported bits:

bit 0: Operation Complete (OP)

bit 1: Not Used

bit 2: Query Error

bit 3: Device Dependent Error

bit 4: Execution Error

bit 5: Command Error

bit 6: Not Used

bit 7: Power On

***IDN?**

Title: Identify

Description: This command returns the following information in <string> format separated by commas: manufacturer name ("Anritsu"), model number, serial number, and firmware package number.

***OPC**
***OPC?**

Title: Operation Complete

Description: The *OPC command causes the device to set the OPC bit of the Standard Event Status Register on the next transition of the No Operation Pending flag from false to true.

The *OPC? command returns 1 in the response whenever the No Operation Pending Flag is true, blocking until the operation is complete.

Note that for IQ streaming, *OPC? will return 1 immediately so that ABORt is not blocked.

***OPT?**

Title: Option Identification

Description: The *OPT command queries the device to identify reportable device options. The response is a list of numeric options separated by commas. If no options are available, a value of "0" will be reported

***RCL <numeric_value>**

Title: Recall Setup

Description: Recalls the saved setup from the desired slot, 1-9. If no setup exists at that slot, will produce an error.

Recalling from slot 0 will recall the instrument's last auto-saved setup.

Parameter(s): <numeric_value>

***RST**

Title: Reset

Description: This command sets the instrument into a known state that is ready for SCPI command processing. Setting the instrument into a known state includes the following steps:

1. All applications beside the default application of the particular model will be turned off.

2. Settings for all applications (regardless of an application being active) are preset to default values.

3. System settings excluding settings associated with client connections (like ethernet addresses or DNS) are preset to default values.

4. Pending operations are aborted and the pending operation flag will be reset. The instrument will be forced into the operation complete command idle state (OCIS) and the operation complete query idle state (OQIS) before the pending operation flag is reset.

***SAV <numeric_value>**

Title: Save Setup

Description: Saves the current instrument configuration to internal storage. Differs from :MMEMory:STORe:STATe in that the setup file is not stored in a user-accessible location.

The command accepts a number 0-9, where numbers 1-9 will save a setup to that numbered slot, and *RCL <n> will recall the setup from that slot.

Entering 0 will instead manually trigger the instrument's auto-save functionality. The instrument saves settings every 5 minutes and when shut down safely, but this may be used for situations where you wish to have a guaranteed save.

The saved setups in slots 1-9 are deleted during a User File Reset and Reset All.

Parameter(s): <numeric_value>

***SRE <numeric_value>**
***SRE?**

Title: Service Request Enable

Description: This command provides access to the Service Request Enable Register.

Refer to the Status System Reporting section within this manual or IEEE 488.2, section 11.5.1 for more information about each supported bit.

Refer to *STB? for the list of supported mask bit values.

Bit 6 (MSS) cannot be set as per IEEE 488.2 standard.

Query returns a NR1 Numeric response.

Parameter(s): <numeric_value>

***STB?**

Title: Status Byte Query

Description: This command queries the Status Byte Register.

Refer to the Status System Reporting section within this manual or IEEE 488.2, section 11.5.1 for more information about each supported bit.

Query returns a NR1 Numeric response.

Supported bits:

bit 0: Not Used

bit 1: Not Used

bit 2: Error Queue (ERRQ)

bit 3: Not Used

bit 4: Message Available (MAV)

bit 5: Standard Event Status (ESB)

bit 6: Master Summary Status (MSS)

bit 7: Not Used

***TST?**

Title: Self Test

Description: Causes the unit to perform a full internal self-test.

Status messages that indicate self-test results are placed in the error queue in the order they occur. Bits in the status register are also affected.

Returns the number of errors placed in the error queue. 0 means the unit passed self-test.

***WAI**

Title: Wait-to-Continue Command

Description: This command causes the device to execute no further commands or queries until the No Operation Pending flag is TRUE.

## 2-2 System Commands

**:DIAGnostic:MEMory?**

Title: Used Memory

Description: This command returns the total memory available on the instrument, the free memory, and the memory used by the backend process in kB.

If the instrument runs into an error while retrieving system memory data, "nan,nan,nan" will be returned as the response and a "-100" (SCPI command error) will be placed in the SCPI error queue.

**:DISPlay:LANGuage <EN|ES|ZH|JA|FR|KO>**
**:DISPlay:LANGuage?**

Title: System Display Language

Description: This command sets and queries the system's displayed language

Parameter(s): <EN | ES | ZH | JA | FR | KO>

Query Return: EN | ES | ZH | JA | FR | KO

Default Value: EN

**:FETCh:GPS?**

Title: Basic GPS Information

Description: This command returns the timestamp, latitude, and longitude of the device. The response is a comma-delimited ASCII response of one of the following forms: NO FIX or GOOD FIX,<timestamp>,<latitude>,<longitude>

If no GPS fix is currently available, the first response form (NO FIX) is returned.

If the GPS does have a fix, the second response form (GOOD FIX) is returned.

<timestamp> is in ISO8601 format. The timestamp provides the 24-hour time, and will include the year/date and/or UTC offset if the hardware supports it. If no UTC offset is provided, the time is in UTC time.

<latitude> and <longitude> are specified in decimal degrees.

**:FETCh:GPS:ALL?**

Title: Full GPS Information + Satellite System Information

Description: This command returns the fix timestamp, latitude, longitude, altitude and information on the satellites used for the last fix and the GNSS systems they are from. The response is in JSON format in the one of the following forms: {"fix":"GPS OFF"}

or {"fix":"GOOD FIX","timestamp":<timestamp>,"latitude":<latitude>,"longitude":<longitude>,"altitude":<altitude>, "satellitesUsed":<satellitesUsed>,"satellites":[ {"name":"GPS","count":x}, {"name":"Galileo","count":x}, {"name":"GLONASS","count":x}, {"name":"BeiDou","count":x}]} {"name":"SBAS","count":x}]} <timestamp> is in ISO8601 format. The timestamp provides the 24-hour time, and will include the year/date and/or UTC offset if the hardware supports it. If no UTC offset is provided, the time is in UTC time.

<latitude> and <longitude> are specified in decimal degrees.

<altitude> specifies the current altitude relative to mean sea level, in meters\.

All satellite systems will be present with a count of 0 if they are not used in the fix.

**:FETCh:GPS:FULL?**

Title: Full GPS Information

Description: This command returns the timestamp, latitude, longitude, altitude, and satellite count of the device. The response is a comma-delimited ASCII response of one of the following forms: NO FIX or GOOD FIX,<timestamp>,<latitude>,<longitude>,<altitude>,<satellites>

If no GPS fix is currently available, the first response form (NO FIX) is returned.

If the GPS does have a fix, the second response form (GOOD FIX) is returned.

<timestamp> is in ISO8601 format. The timestamp provides the 24-hour time, and will include the year/date and/or UTC offset if the hardware supports it. If no UTC offset is provided, the time is in UTC time.

<latitude> and <longitude> are specified in decimal degrees.

<altitude> specifies the current altitude relative to mean sea level, in meters.

<satellites> specifies an integer count of the number of satellites currently used in the fix.

**:FETCh:GPS:LAST?**

Title: Last GPS Fix

Description: This command returns the timestamp, latitude, longitude, and altitude of the last fixed GPS result. The response is a comma-delimited ASCII response of one of the following forms: NO FIX or GOOD FIX,<timestamp>,<latitude>,<longitude>,<altitude> If a GPS fix has never been acquired, the first response form (NO FIX) is returned.

If a GPS fix was previously acquired, the second response form (GOOD FIX) is returned.

<timestamp> is in ISO8601 format. The timestamp provides the 24-hour time, and will include the year/date and/or UTC offset if the hardware supports it. If no UTC offset is provided, the time is in UTC time.

<latitude> and <longitude> are specified in decimal degrees.

<altitude> specifies the current altitude relative to mean sea level, in meters.

**:INSTrument:APPLication:STATe <SPA>,<0 | 1 | ON | OFF>**

Title: Activate or Deactivate Application

Description: This command turns the specified application ON or OFF one at a time. When an application is turned ON (i.e. activated), it is loaded into memory and can be used to make measurements. An application must be activated before it will respond to commands. Please expect some delay in the application response to a command right after being turned on. The delay will vary depending on the application type and the options being installed. When an application is turned OFF (i.e. deactivated), any measurements it is making are stopped and it is unloaded from memory. Attempting to turn ON the same application twice or turn OFF an application that is not currently ON is not permitted and will result in an execution error (-200).

Attempting to turn on a different application (not the currently active application) requires turning the current application OFF. Otherwise, it will result in an execution error. To turn off an application uses the :INSTrument:APPLication:STATe command.

Unlike INSTrument:SELect, which selects a given application and makes it active if it is not already active, this command can be used to activate an application without selecting it.

If the requested application is the selected application (which can be queried with INSTrument:SELect?), turning it OFF will cause the current application selection to be reported as NONE via INSTrument:SELect?.

This command takes two parameters. The first parameter is a string value that specifies which application should be turned ON or OFF. The permitted values of this parameter are the same as those used by the INSTrument:SELect command. The second parameter is a Boolean value that specifies whether the application should be turned ON or OFF.

After a reset, the default selected application is the only active application.

Parameter(s): <SPA>,<0 | 1 | ON | OFF>

**:INSTrument:CATalog:ACTive?**

Title: List Active Applications

Description: This command queries all active applications. The response is returned as a comma delimited list of application names. The application names are the same as those returned by the INSTrument:CATalog:FULL? query or used as parameters to the INSTrument:SELect command.

If no applications are currently active, this query will return NONE.

**:INSTrument:CATalog:ALL?**

Title: Query Available Applications with all information

Description: This query returns a JSON list of avaialble applications with metadata including active status, appId modes and SCPI API version. [ {"name":"SPA", "active":"true", "appId":"1","modes":["SPEC","NRAD"], "version": 1}, {"name":"VNA", "active":"false", "appId":"26","modes":["VNA"], "version": 1}, {"name":"SYSTEM", "version":1 } ]

The list will always contain at least one item since the System application will always be included.

**:INSTrument:CATalog:FULL?**

Title: Query Available Applications

Description: This query returns a list of string - number pairs. The string contains the name of the available application. The immediately following NR1-formatted number is its associated application number. All response data elements are comma separated.

If no application is available, a null string followed by a zero is returned.

**:INSTrument:NSELect <numeric_value>**
**:INSTrument:NSELect?**

Title: Select Application by Number

Description: Sets the selected application based on the value of <integer>. The query version returns the number associated with the current application. The list of valid integers and the applications they correspond to can be queried with INSTrument:CATalog:FULL?

Selecting a different application (not the currently active application) requires turning the current application OFF. Otherwise, it will result in an execution error. To turn off an application uses the :INSTrument:APPLication:STATe command.

If no applications are active, the query version of this command will return 0.

Parameter(s): <numeric_value>

**:INSTrument[:SELect] <SPA>**
**:INSTrument[:SELect]?**

Title: Select Application by Name

Description: Sets the selected application based on the application name specified by <identifier>. The <identifier> is a valid application name (character data). The list of valid application names can be queried with INSTrument:CATalog:FULL?

Selecting a different application (not the currently active application) requires turning the current application OFF. Otherwise, it will result in an execution error. To turn off an application uses the :INSTrument:APPLication:STATe command.

The query version returns the name of the current application.

If no applications are active, the query version of this command will return NONE.

Parameter(s): <SPA>

Query Return: NONE|SPA

Default Value: NONE

### :MMEMory:CATalog:DIRectory? <string>,<string>

Title: Memory Catalog Directory

Description: Returns the non-recursive contents of the memory specified by the MSUS and DIRECTORY parameters. Both parameters are case sensitive.

Parameter 1: Directory. Use "/" as a directory separator. Note that this parameter should NOT be an empty string. To access the root directory, pass in "/" as the first parameter.

Parameter 2: MSUS (Use :MMEMory:CATalog:MSUSs query command to retrieve the list of available storage devices).

For example:

MMEMory:CATalog:DIRectory? "/", "USB1-1"

The response is formatted as follows: <total space used on MSUS>,<total space available on MSUS>,{<file entry1>},..{<file entryN>} where <file entry> is: <file or directory name>,<file type>,<file size>

Parameter(s): <string>,<string>

### :MMEMory:CATalog:DIRectory:SIZE? <string>,<string>

Title: MMemory Catalog Directory Size

Description: This command is query-only and returns the size of a given directory. It has two parameters. Both are case sensitive.

<directory>,<msus>

Parameter 1: The full path of a directory to query the size. Use "/" as a directory separator. Note that this parameter should NOT be an empty string. To access the root directory, pass in "/" as the first parameter.

Parameter 2: MSUS (Use :MMEMory:CATalog:MSUSs query command to retrieve the list of available storage devices).

For example:

MMEMory:CATalog:DIRectory:SIZE? "/", "USB1-1"

The response is a single ASCII string of the size of the given directory.

Parameter(s): <string>,<string>

### :MMEMory:CATalog:MSUSs?

Title: Memory Catalog Msuss

Description: Returns a list of all available mass storage devices present, formatted as follows: <Device Name 1>,<Device Name 2>,...<Device Name N>

For example, sending:

MMEMory:CATalog:MSUSs?

could return:

Internal,USB1-1

**:MMEMory:CDIRectory <string>**
**:MMEMory:CDIRectory?**

Title: Default Mass Storage Directory

Description: This command selects the default directory on the default mass storage device (see :MMEMory:MSIS) for use with MMEMory commands. The parameter is case sensitive. Use "/" as a directory separator.

For example:

MMEMory:CDIRectory "/example_folder"

The set command will fail and an error is reported if the requested directory does not exist. Note that the query command returns the name of the default directory that was previously set and the device does not check whether the directory is still present.

Parameter(s): <string>

Default Value: /

**:MMEMory:COPY <string>,<string>,<string>,<string>**

Title: Copy File

Description: This command copies the specified source file into a new file located at the specified destination location. Four parameters are required for this command:

1. Source File Location: The file path to the file to be copied.

2. Source MSUS: The mass storage device the file is located in (i.e. Internal).

3. Destination File Location: The destination file path that the file should be copied to.

4. Destination MSUS: The mass storage device that the destination file should be written to (i.e. Internal).

For example:

MMEMory:COPY "/example_file.ext", "Internal", "/example_file.ext", "USB1-1"

Please note that if a file already exists at the destination location or if the source file does not exist, the command will fail to execute and add an execution error into the SCPI error queue. This command will only copy files. If a directory path is passed in as a parameter, the command will fail to execute.

Parameter(s): <string>,<string>,<string>,<string>

### :MMEMory:CREate:DIRectory <string>,<string>

Title: Create Directory

Description: Creates a directory at the specified mass storage device. Both parameters are case sensitive.

Parameter 1: Directory. Use "/" as a directory separator.

Parameter 2: MSUS (Use :MMEMory:CATalog:MSUSs query command to retrieve the list of available storage devices)

For example:

MMEMory:CREate:DIRectory "/example_folder", "USB1-1"

If the intermediate directories in the path don't exist, the command will automatically create them. Common causes of the command failure are:

- The MSUS device is not present (Use :MMEMory:CATalog:MSUSs query command to retrieve the list of available storage devices)

- The total space available on MSUS device is not big enough (Use :MMEMory:CATalog:DIRectory query command to retrieve the total space available on MSUS)

Parameter(s): <string>,<string>

**:MMEMory:DATA <string>,<string>,<block data>**
**:MMEMory:DATA? <string>,<string>**

Title: File Transfer

Description: This command imports/exports a file to/from the instrument. Data is transferred to/from the instrument as an IEEE definite length arbitrary block response, which has the form <header><block>.

This set command takes three parameters:

1. File Path: The path of the file to be written

2. Msus: The mass storage device to write the file to

3. Block Data: The data to be written to the instrument in block data format

If any directories in the file path do not exist, the instrument will automatically create the required directories. Please note that the maximum file transfer size to the instrument is 25 MB.

The ASCII header specifies the number of data bytes of the file. It looks like #AX, where A is the number of digits in X and X is the number of bytes in the <block>. The first character is the literal ascii hash '#' 043. The second character (A) is a single ascii digit '1' to '9' describing the number of bytes in the length section (X). This number is called nlength. The next nlength bytes make up an ascii string of digits '1' to '9' describing the length of the <block> data.

For example, if the first 6 bytes are #49999, then the nlength is 4. The 4 bytes of length are 9999. After that follows the <block>, which would be 9999 bytes in size.

The query command takes two parameters.

1. File Path: The path to the file to be retrieved.

2. Msus: The mass storage device to retrieve the file from.

The file is returned in block data format with an ASCII header The query command will return a #10 if the file cannot be found.

Set Command Parameter(s): <string>,<string>,<block data>

Query Command Parameter(s): <string>,<string>

**:MMEMory:DELete:DIRectory <string>,<string>**

Title: Delete Directory

Description: Deletes the specified directory and all its content at the specified mass storage device. The two parameters for this command are directory name and mass storage device. Both parameters are case sensitive. Use "/" as a directory separator. To delete the root folder in a specified mass storage device, enter "/" for directory name parameter.

For example:

MMEMory:DELete:DIRectory "/example_folder", "USB1-1"

The available mass storage devices can be retrieved by using the :MMEMory:CATalog:MSUSs command. The command will fail if the directory doesn't exist or the mass storage device is not present.

Parameter(s): <string>,<string>

### :MMEMory:DELete:FILe <string>,<string>

Title: Delete File

Description: Deletes the specified file at the specified mass storage device. The two parameters for this command are file name with relative path and mass storage device. Both parameters are case sensitive. Use "/" as a directory separator.

For example:

MMEMory:DELete:FILe "/example_folder/example_file.ext", "USB1-1"

The available mass storage devices can be retrieved by the :MMEMory:CATalog:MSUSs command. The command will fail if the file doesn't exist or the mass storage device is not present.

Parameter(s): <string>,<string>

### :MMEMory:EJECt <string>

Title: Eject USB drive

Description: This command safely ejects (unmounts) a USB drive from the instrument. Following this command, a USB drive can be safely removed from the instrument. See :MMEMory:CATalog:MSUSs for more information on obtaining the name of the USB drive(s) currently mounted on the instrument.

For example:

MMEMory:EJECT "USB1-1"

Parameter(s): <string>

### :MMEMory:LOAD:STATe <numeric_value>,<string>,<string>

Title: Load State

Description: This command recalls the specified setup. The file location is resolved using the MSUS and file path parameters. These parameters are case sensitive.

Parameters:

1. Numeric Value: currently unused. Send a 0.

2. File path: The file path (including file name) relative to the MSUS root directory. File extension is optional.

3. MSUS: Device to recall the file from. See MMEMory:CATalog:MSUSs? for information on obtaining the list of available devices.

For example:

MMEMory:LOAD:STATe 0,"/example_setup_file", "Internal"

The model of the device that the setup was created in must match the model of the device to recall in; otherwise, the recall will be rejected. The options of the device that the setup was created in must be enabled in the device to recall in; otherwise, the recall will be rejected.

Parameter(s): <numeric_value>,<string>,<string>

**:MMEMory:LOAD:TRACe <string>,<string>,<string>**

Title: Load Trace

Description: This command recalls the specified measurement and setup. After recall all traces will have update state turned off, but sweeping will continue with the recalled setup. This means that traces will retain the recalled measurement data until the trace update state is turned on. The file location is resolved using the MSUS and file path parameters. These parameters are case sensitive.

Parameters:

1. Label: Used to specify which trace to recall. Currently, the label only supports recalling all traces. Send an empty string or "ALL" to load all traces.

2. File path: The file path (including file name) relative to the MSUS root directory. File extension is optional.

3. MSUS: Device to recall the file from. See MMEMory:CATalog:MSUSs? for information on obtaining the list of available devices.

For example:

MMEMory:LOAD:TRACe "ALL", "/example_measurement_file", "Internal"

The model of the device that the measurement was saved in must match the model of the device to recall in; otherwise, the recall will be rejected. The options of the device that the measurement was saved in must be enabled in the device to recall in; otherwise, the recall will be rejected.\

Parameter(s): <string>,<string>,<string>

**:MMEMory:MSIS <string>**
**:MMEMory:MSIS?**

Title: Default Mass Storage Device

Description: This command selects the default device for use with MMEMory commands. The mass storage device parameter is case sensitive and must match a device returned from the :MMEMory:CATalog:MSUSs? command

For example:

MMEMory:MSIS "Internal"

If Secure Data option is installed, the query command will return the next available external storage device.

If no external storage device is plugged in, the query command returns "None".

Parameter(s): <string>

Default Value: Internal

### :MMEMory:STORe:STATe <numeric_value>,<string>,<string>

Title: Store State

Description: This command saves the current setup to the specified file location. The file location is resolved using the MSUS and file path parameters. These parameters are case sensitive. If a file with the resolved name already exists, it will be overwritten.

Parameters:

1. Numeric Value: currently unused. Send a 0.

2. File path: The file path (including file name) relative to the MSUS root directory. File extension should not be specified.

3. MSUS: Device to save the file on. See MMEMory:CATalog:MSUSs? for information on obtaining the list of available devices.

For example:

MMEMory:STORe:STATe 0,"/example_setup_file", "Internal"

A common cause of the command failure is not having enough space available on MSUS to save the setup. Use :MMEMory:CATalog:DIRectory query command to retrieve the total space available on MSUS.

This command is disabled if Secure Data Option is available and enabled.

Parameter(s): <numeric_value>,<string>,<string>

### :MMEMory:STORe:TRACe <string>,<string>,<string>

Title: Store Trace

Description: This command saves the current measurement data along with the current setup to the specified file location. The file location is resolved using the MSUS and file path parameters. These parameters are case sensitive. Parameters:

1. Label: Used to specify which trace to save. Currently, the label only supports saving all traces. Send an empty string or "ALL" to store all traces.

2. File path: The file path (including file name) relative to the MSUS root directory. File extension should not be specified.

3. MSUS: Device to save the file on. See MMEMory:CATalog:MSUSs? for information on obtaining the list of available devices.

For example:

MMEMory:STORe:TRACe "ALL", "/example_measurement_file", "Internal"

A common cause of the command failure is not having enough space available on MSUS to save trace. Use :MMEMory:CATalog:DIRectory query command to retrieve the total space available on MSUS.

Parameter(s): <string>,<string>,<string>

**:OUTPut:ROSCillator:STATe <0 | 1 | ON | OFF>**
**:OUTPut:ROSCillator:STATe?**

Title: 10MHz Reference Output State

Description: This command enables or disables the 10MHz reference output. The query returns 1 if 10MHz reference output is enabled; otherwise, returns 0. Not available on the MS2080A or MS2070A.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:BIAS:EXTernal:CURRent?**

Title: External Bias Current

Description: This command queries the actual bias current in A.

**[:SENSe]:BIAS:EXTernal:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:BIAS:EXTernal:STATe?**

Title: External Bias On/Off

Description: This command enables or disables the external bias. The query returns 1 if external bias is enabled; otherwise, returns 0.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:BIAS:EXTernal:TRIPped[:STATe]?**

Title: External Bias Fault

Description: This command queries whether the external bias is tripped. The query returns 1 for tripped; otherwise, returns 0.

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:BIAS:EXTernal:TRIPped[:STATe]:CLEar**

Title: Clear External Bias Fault

Description: This command clears the external bias tripped status. See command [:SENSe]:BIAS:EXTernal:TRIPped[:STATe] for retrieving the current tripped status.

**[:SENSe]:BIAS:EXTernal:VOLTage <numeric_value>**
**[:SENSe]:BIAS:EXTernal:VOLTage?**

Title: External Bias Voltage

Description: This command sets the desired voltage for the external bias. The query returns the actual bias voltage in V.

Parameter(s): <numeric_value>

**[:SENSe]:FREQuency:REFerence:SOURce?**

Title: Frequency Reference Source

Description: Returns the current frequency reference source used by the instrument, as specified in the following table:

Return value: Description

GPS: GPS High Accuracy

ACQ: Acquiring GPS Reference

INT: Internal Standard Accuracy

HI: Internal High Accuracy

EXT: External (10 MHz) Reference

Query Return: ACQ | EXT | GPS | HI | INT

Default Value: INT

**[:SENSe]:RF:SAFE:MODE <0 | 1 | ON | OFF>**
**[:SENSe]:RF:SAFE:MODE?**

Title: RF Safe Mode

Description: This command enable or disable RF safe mode state

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:TIME:REFerence:SOURce?**

Title: Time Reference Source

Description: Returns the current time reference used by the instrument, as specified in the following table:

Return value Description

GPS GPS High Accuracy

INT Internal Standard Accuracy

HI Internal High Accuracy

Query Return: GPS | HI | INT

Default Value: INT

**:SYSTem:COMMunicate:LAN:CONFig <string>,[<string>],[<string>]**
**:SYSTem:COMMunicate:LAN:CONFig?**

Title: Static LAN Configuration

Description: This command set and queries the static ethernet configuration of the device. The static configuration allows user to specify the ip, gateway, and subnet mask of the unit on a network. Parameters:

- Static IP Address: The desired IP address of the unit.

- Gateway: (optional)The network gateway

- Subnet Mask: (optional) The subnet mask of the network the device is connected to.

For a subnet mask of 255.255.255.0(/24), you only have a valid host IP range of x.x.x.1-x.x.x.254. The first IP in the subnet (x.x.x.0) is reserved for the network ID and the last one (x.x.x.255) is reserved as the broadcast ID. You can have a different range of valid host addresses if you change the subnet mask to a different number. For example, if a subnet mask of 255.255.254.0 is used, the gateway can be set to 192.168.101.0 since the valid host IP range is 192.168.100.1-192.168.101.254.

More information on subnetting can be found here:

https://docs.microsoft.com/en-us/troubleshoot/windows-client\networking/tcpip-addressing-and-subnetting

A calculator for finding valid host IP addresses in a subnet can be found here:

https://www.calculator.net/ip-subnet-calculator.html

The Gateway and Subnet Mask settings will remain no change if user has no input.

The dhcp will be turned off.

A new valid configuration will automatically be applied to the device. The user will be required to access the unit through the new configuration.

CAUTION: Consult with your network administrator when configuring the network interface to avoid potential loss of access or discovery of the device.

Parameter(s): <string>,[<string>],[<string>]

**:SYSTem:COMMunicate:LAN:CONFig:CURRent?**

Title: Current LAN Configuration

Description: This command queries the current ethernet configuration of the device.

**:SYSTem:COMMunicate:LAN:DHCP <ON|OFF>**
**:SYSTem:COMMunicate:LAN:DHCP?**

Title: DHCP Configuration

Description: This command sets and queries the DHCP configuration of the device. If the DHCP configuration is set to OFF, the device is configured to the static ethernet configuration (See :SYSTem:COMMunicate:LAN:CONFig). If the DHCP configuration is set to ON, the device will obtain its IP address, gateway, and subnet mask from the DHCP server in the network.

This set command should be used to caution, as changing the ethernet configuration will result in temporary loss of communication with the device.

CAUTION: Consult with your network administrator when configuring the network interface to avoid potential loss of access or discovery of the device.

Parameter(s): <ON|OFF>

Query Return: ON|OFF

Default Value: OFF

**:SYSTem:COMMunicate:LAN:DNS <string>,<string>,<string>**
**:SYSTem:COMMunicate:LAN:DNS?**

Title: Dynamic DNS Configuration

Description: This command sets and queries the Dynamic Domain Name Server (Dynamic DNS) configuration of the device. Currently, only http://www.noip.com (TM) is available for use. For support, please contact the vendor at http://www.noip.com. Parameters:

- Dynamic DNS NoIp.com Host Name: Desired host name of the device.

- Dynamic DNS NoIp.com Username: noip.com username.

- Dynamic DNS NoIp.com Password: noip.com password.

Parameter(s): <string>,<string>,<string>

**:SYSTem:COMMunicate:LAN:DNSServer:APPend <string>, {<string>}, ...**

Title: Domain Name System Server Append

Description: This command will add one or more Domain Name System Servers to the end of the device's current list of DNS Servers.

To view the entire list of DNS Servers, see SYSTem:COMMunicate:LAN:DNSServer:LIST. To remove a DNS Server, see SYSTem:COMMunicate:LAN:DNSServer:DELete.

This command takes a list of one or more string values. For example: SYSTem:COMMunicate:LAN:DNSServer:APPend "8.8.8.8", "8.8.4.4"

Parameter(s): <string>, {<string>}, ...

**:SYSTem:COMMunicate:LAN:DNSServer:DELete <string>, {<string>}, ...**

Title: Domain Name System Server Delete

Description: This command will delete the device's given Domain Name System Servers from the device's current list of DNS servers. To view the entire list of DNS Servers, see SYSTem:COMMunicate:LAN:DNSServer:LIST.

This command takes a list of one or more string values. For example: SYSTem:COMMunicate:LAN:DNSServer:DELete "8.8.8.8", "8.8.4.4"

Parameter(s): <string>, {<string>}, ...

**:SYSTem:COMMunicate:LAN:DNSServer:LIST <string>, {<string>}, ...**
**:SYSTem:COMMunicate:LAN:DNSServer:LIST?**

Title: Domain Name System Server List

Description: This command sets and retrieves a list of the device's currently set Domain Name System Servers.

The DNS Server is used to resolve a Domain Name. The list of DNS Servers will be contacted after any DNS Server assigned by DHCP Setting, if applicable, and will be contacted in the order given in the list.

This command takes a list of one or more string values. Setting the DNS server list with this command will override the currently set DNS server list.

An example of using the command: SYSTem:COMMunicate:LAN:DNSServer:LIST "8.8.8.8", "8.8.4.4"

This query version of this command returns a comma delimited list of strings. For example: 8.8.8.8, 8.8.4.4

To remove DNS Server, see SYSTem:COMMunicate:LAN:DNSServer:DELete.

To add to the end of the current list, see SYSTem:COMMunicate:LAN:DNSServer:APPend.

Parameter(s): <string>, {<string>}, ...

**:SYSTem:COMMunicate:LAN:FTP:STATe <0 | 1 | ON | OFF>**
**:SYSTem:COMMunicate:LAN:FTP:STATe?**

Title: FTP Server State

Description: This command sets the system up to execute (ON) or bypass (OFF) the FTP server startup on the next instrument boot.

The query returns the state of the FTP server based on the bypass flag, 0 for bypass, 1 for start. The actual state of the FTP server is not check for the query.

Parameter(s): <0 | 1 | ON | OFF>

**:SYSTem:COMMunicate:LAN:HOSTname <string>**
**:SYSTem:COMMunicate:LAN:HOSTname?**

Title: Local Host Name

Description: This command sets and retrieves the local host name of the instrument. A valid hostname may contain only the ASCII letters 'a' through 'z' (in a case-insensitive manner), the digits '0' through '9', and the hyphen ('-'). They cannot start/end with '-'.

No other symbols, punctuation characters, or white space are permitted.

Parameter(s): <string>

**:SYSTem:COMMunicate:LAN:HTTP:SECure:STATe <0 | 1 | ON | OFF>**
**:SYSTem:COMMunicate:LAN:HTTP:SECure:STATe?**

Title: HTTPS Port State

Description: This command sets the system up to allow (ON) or bypass (OFF) the HTTPS port setup on the next instrument boot.

The query returns the state of the HTTPS port based on the bypass flag, 0 for bypass, 1 for configure. The actual state of the HTTPS port is not check for the query. Bypassing the HTTPS port will prevent use of remote software file transfer.

Parameter(s): <0 | 1 | ON | OFF>

**:SYSTem:COMMunicate:LAN:SECure:CREDentials <string>,<string>,<RSA>**

Title: Secure Communication Custom Certificate Key Pair

Description: Loads a custom certificate and key pair into the instrument to be used with secure connections.

The command takes 3 parameters:

1. Certificate contents (in PEM format).

2. Private key contents (in PEM format).

3. Key type. Currently, only RSA is supported.

The command is only available if the Secure Communication option is installed.

On a secure connection, the command is only accessible after command :SYSTem:PASSword:CENable has been received with a valid password.

Parameter(s): <string>,<string>,<RSA>

**:SYSTem:COMMunicate:LAN:SECure:SSLCertificate?**

Title: Secure Communication Certificate Information

Description: Returns the information about the SSL certificate being used in the secure connection. The information is returned in JSON format and consists of the following fields:

1. Start Date("notBefore"): When the certificate was generated (in GMT).

2. End Date("notAfter"): When the certificate will expire (in GMT).

3. Country("C"): Country of origin.

4. State("ST"): State of origin.

5. Location("L"): City of origin.

6. Organization("O"): Organization of origin.

7. Common Name("CN"): Associated host name.

All fields will be populated into a JSON array. The resulting response will be similar to the following: [{"C":"<country>", "ST":"<state>", "L":"<city>", "O":"<organization>", "CN":"<host_name>", "notAfter":"<notAfter_date>", "notBefore":"<notBefore_date>"}]

The command is only available if the Secure Communication option is installed.

**:SYSTem:COMMunicate:WLAN:CONFig:CURRent?**

Title: Current WLAN Configuration

Description: This command queries the device wireless LAN configuration.

For example: { "ipAddress" : "192.168.0.190",

"gateWay" : "192.168.0.1",

"subnetMask" : "255.255.255.0"

}

**:SYSTem:COMMunicate:WLAN:CONNection <string>,[<string>]**

Title: Connect to a WiFi Access Point

Description: This command connects to a WiFi access point. The first string parameter is the ssid of the access point. The second string parameter, which is optional is the password of the access point.

Parameter(s): <string>,[<string>]

**:SYSTem:COMMunicate:WLAN:CONNection:ACTive?**

Title: WiFi Active Connection Information

Description: This command returns properties of the current WLAN connection.

If SYSTem:COMMunicate:WLAN:CONNection:STATus is not CONNected, the response is an ASCII empty JSON object: {} Otherwise the response is an ASCII JSON object with the following property:

SSID (ssid): The Service Set Identifier identifying the access point connected to.

For example: { "ssid" : "ExampleAccessPoint",

}

**:SYSTem:COMMunicate:WLAN:CONNection:DISConnect**

Title: Disconnect from a WiFi Access Point

Description: This command disconnects from the currently connected WiFi access point. This will also turn off auto connecting to another wireless network until the device is rebooted or WiFi is toggled on and off.

**:SYSTem:COMMunicate:WLAN:CONNection:FORGet <string>**

Title: Forget a WiFi Access Point

Description: This command disconnects and forgets a WiFi access point.

The first string parameter is the ssid of the access point.

Parameter(s): <string>

**:SYSTem:COMMunicate:WLAN:CONNection:FORGet:ALL**

Title: Forget All WiFi Access Points

Description: This command disconnects and forgets all saved WiFi Access Points

**:SYSTem:COMMunicate:WLAN:CONNection:SAVed?**

Title: Saved WiFi Access Points

Description: This command returns all saved WiFi access points as JSON array

For example: [

"TheBestNetwork",

"TheShoppingMall",

]

**:SYSTem:COMMunicate:WLAN:CONNection:STATus?**

Title: WiFi Connection Status

Description: This command returns the connection status of the WLAN device. The response is ASCII character data of one of the following values:

Parameter value Description

CONNected A connection is established

DISConnected No connection currently exists

Query Return: CONN | DISC

Default Value: DISConnected

### :SYSTem:COMMunicate:WLAN:NETWorks?

Title: WiFi Access Points

Description: This command returns available WiFi access points.

If SYSTem:COMMunicate:WLAN:STATe is false, the response is an empty list [] Otherwise the response is ASCII of JSON object array with the following properties:

1. SSID (ssid): The Service Set Identifier identifying the access point connected to.

2. Frequency (frequency): The radio channel frequency in use by the access point, in MHz.

3. Max Bit Rate (rate): The maximum bitrate this access point is capable of.

4. Signal Strength (signal): The current signal quality of the access point, in percent.

5. BSSID (bssid): The hardware address of the access point.

6 Security Methods (security) : the security methods supported by the access point

For example: {

"ssid" : "RAPTOR",

"frequency" : "5220 MHz",

"rate" : "54 Mbit/s",

"signal" : 72,

"bssid" : "00:13:1A:34:D8:52",

"security" : "WPA1","WPA2"

},

{

"ssid" : "AM-Employee",

"frequency" : "5 MHz",

"rate" : "5.22",

"signal" : 62,

"bssid" : "06:18:1A:31:4E:62"

"security" : "WPA1","WPA2"

}

### :SYSTem:COMMunicate:WLAN:NETWorks:REFResh

Title: Refresh WiFi Access Points

Description: This command will force the wireless network card to rescan local access points. This command will not return any data. Scanning for access points can take between 10-40 seconds before updating. Network rescans are rate limited by the hardware, consecutive calls to this command will not result in multiple rescans by the hardware. Calling SYST:COMM:WLAN:NETW? too soon after calling this command can result in a return of non updated data, or an empty list.\

**:SYSTem:COMMunicate:WLAN:STATe <0 | 1 | ON | OFF>**
**:SYSTem:COMMunicate:WLAN:STATe?**

Title: Enable or Disable WiFi

Description: This command toggles the WLAN interface on or off.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:SYSTem:DATE <numeric_value>,<numeric_value>,<numeric_value>**
**:SYSTem:DATE?**

Title: System Date

Description: This command sets and queries the system's internal calendar. Changing the system date does not affect the system time zone (if time zone is available). The three parameters for this command are <year>,<month> and <day>. The query response message shall consist of three fields separated by commas: <year>,<month>,<day>.

The year shall be entered as a four-digit number, including century and millennium information. This shall not be affected by a *RST command. Note: Some products only support the query version of this command.

Parameter(s): <numeric_value>,<numeric_value>,<numeric_value>

**:SYSTem:DEFault:RESet:ALL**

Title: Default Reset All

Description: This command performs a factory reset as well as deletes last saved settings, log files, and user files. The factory reset includes presetting parameters in all applications, presetting all system settings with the exception of ethernet settings (DHCP On/Off, static IP, Static Gateway, Static Subnet) and time zone settings, and cleaning up all previous access point connections if the instrument is equipped with WiFi.

**:SYSTem:DEFault:RESet:DATA <USER|SYSTem>**

Title: Reset System Files

Description: This command deletes the instrument data files as specified in the following table:

Parameter Value Description

USER Deletes all user files in the instrument's internal memory including measurements, setup files, and screen shots. User-customized system files will not be deleted.

SYSTem Deletes all user-customized system files including keyboard EZ names, cable lists, antenna lists, and log files.

Parameter(s): <USER|SYSTem>

**:SYSTem:DEFault:RESet:FACTory**

Title: Factory Default Reset

Description: This command presets parameters in all applications as well as system settings with the exception of ethernet settings (DHCP On/Off, static IP, Static Gateway, Static Subnet) and time zone setting to their factory default values, and restarts the instrument. Last saved settings will be deleted. If unit equips with WiFi, the previous access point connections will be clean up.

**:SYSTem:ERRor:DISPlay <0 | 1 | ON | OFF>**
**:SYSTem:ERRor:DISPlay?**

Title: SCPI Error Message State

Description: Sets or queries the state of SCPI error messages being displayed. By default, when a SCPI error enters the error queue it will be displayed on-screen. This command toggles whether those errors should be displayed. This command defaults to 0, meaning errors will NOT be displayed.

Usage: Suppress SCPI Error Messages: SYST:ERR:DISP 0 Show all SCPI Error Messages: SYST:ERR:DISP 1.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:SYSTem:ERRor[:NEXT]?**

Title: System Error Queue

Description: If an error occurs, the error number and message are placed in the error queue, which can be read by this query command. Errors are cleared by reading them. Error code 0 is "No error".

Error codes from -100 to -199 belong to the Command error category and set bit 5 of the standard ESR register.

Error codes from -200 to -299 belong to the Command error category and set bit 4 of the standard ESR register.

Error codes from -400 to -499 belong to the Query error category and set bit 2 of the standard ESR register.

Error codes from -300 to -399 and 1 to 32767 belong to the Device-specific error category and set bit 3 of the standard ESR register.

Negative error numbers (command error, execution error, device-dependent error, query error) are standard SCPI errors. Positive error numbers are device specific errors, not standard SCPI errors.

The error queue is also cleared by *CLS, *RST, and when power is turned on.

If more errors have occurred than can fit in the buffer, the last error stored in the queue (the most recent error) is replaced with -350, Queue overflow. No additional errors are stored until removing errors from the queue. If no errors have occurred when reading the error queue, the instrument responds with 0, No error.

### :SYSTem:FIRMware:UPDate <string>,<string>

Title: Firmware Update

Description: Initiate a firmware update from the MSUS whose path is specified as parameters. If the requested firmware package is not recognized, an appropriate error will be added to the error queue. Otherwise, the instrument will verify that the package is safe to install, update to the new firmware.

The two parameters for this command are file name with relative path and mass storage device. Both parameters are case sensitive. Use "/" as a directory separator.

For example:

SYSTem:FIRMware:UPDate "V2019.01.1.raucb", "USB4-1"

The available mass storage devices can be retrieved by the :MMEMory:CATalog:MSUSs command. The command will fail if the file doesn't exist or the mass storage device is not present.

Parameter(s): <string>,<string>

### :SYSTem:FIRMware:UPDate:LIST?

Title: Firmware Package List

Description: This command queries the instrument for a list of firmware packages that can be installed from mounted USB block devices. Please note that firmware packages are only detected on the top level of a USB block device. Any files located in subfolders will be ignored.

The response is in JSON format and consists of the path of the package relative to the root of the MSUS, the MSUS that the package is located in, and the version of the package.

The following is an example of the response:
[{"msus":"USB1-1","path":"V2018.2.22.raucb","version":"V2018.2.22"},
{"msus":"USB2-1","path":"V2018.2.19.raucb","version":"V2018.2.19"}]

These package names enumerate the valid parameter values to the :SYSTem:FIRMware:UPDate command.

### :SYSTem:FIRMware:UPDate:REMote <string>

Title: Remote Firmware Update

Description: Initiate a remote firmware update to the package whose name is specified as a parameter. Use :SYSTem:FIRMware:UPDate:REMote:LIST? to inspect the packages that can be installed. If the requested firmware package is not recognized, an appropriate error will be added to the error. Otherwise, the instrument will download the specified firmware package, verify that the package is safe to install, update to the new firmware.

The instrument will disconnect all remote clients and reboot after the new firmware has been installed.

If the <STRING DATA> argument begins with (case insensitive) any of

http://

https://

ftp://

Then the firmware update will be downloaded from that URL. The URL must be percent-encoded per RFC-2396 section 2.4.1.

Parameter(s): <string>

**:SYSTem:FIRMware:UPDate:REMote:AUTO <0 | 1 | ON | OFF>**
**:SYSTem:FIRMware:UPDate:REMote:AUTO?**

Title: Remote Firmware Update Auto Check

Description: This command sets and queries the state to enable automatically checking a remote server (see SYSTem:FIRMware:UPDate:REMote:SOURce?) for software updates. If this command is set to ON, The server will be checked for a newer version on bootup and every 24 hours thereafter. If this command is set to OFF, no automatic checking will take place, but a check can be initiated manually (see SYSTem:FIRMware:UPDate:REMote:LIST:REFResh command). If the check finds a newer version available, the SYSTem:FIRMware:UPDate:REMote:LATest? query will indicate the newer version. Use the SYSTem:FIRMware:UPDate:REMote command to initiate the download and install.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**:SYSTem:FIRMware:UPDate:REMote:LATest?**

Title: Check For Firmware Update

Description: Check whether the instrument firmware is at the latest version. This command will access the package list specified by SYST:FIRM:UPD:REM:SOUR and return a response of the form <string>,<number> where <string> is the name of the latest firmware package, and <number> is 1 if the latest firmware package is newer than the version of firmware currently installed (a firmware update is available), or 0 if the current instrument firmware version matches the latest version (the instrument is up to date).

Clients can update the instrument to the latest firmware by passing the <string> component of the response as a parameter to the SYST:FIRM:UPD:REM command.

**:SYSTem:FIRMware:UPDate:REMote:LIST?**

Title: Remote Firmware Package List

Description: This command queries the instrument for a list of firmware packages that can be installed remotely. The package list will be populated from a resource file that is downloaded from a URL specified by the current value of :SYSTem:FIRMware:UPDate:REMote:SOURce.

The response is ASCII response consisting of one or more comma-delimited package names. For example: 2.0.0,1.0.0

These package names enumerate the valid parameter values to the :SYSTem:FIRMware:UPDate:REMote command.

### :SYSTem:FIRMware:UPDate:REMote:LIST:REFResh

Title: Remote Firmware Package List Refresh

Description: This command causes the list of firmware packages that can be installed remotely (:SYSTem:FIRMware:UPDate:REMote:LIST?) to be refreshed from the remote source server immediately. Normally the list is automatically updated on a timer interval to limit the amount of network requests. The exact timer interval varies by model and could be anywhere from one minute to one day. Use this command if you do not want to wait for the next automatic timer-based update to the remote firmware package list.

Note that this command only initiates the request to the server to get the updated list of packages. It does not wait until the list (as visible by :SYSTem:FIRMware:UPDate:REMote:LIST?) is actually updated as the request is made asynchronously.

**:SYSTem:FIRMware:UPDate:REMote:SOURce <string>**
**:SYSTem:FIRMware:UPDate:REMote:SOURce?**

Title: Remote Firmware Package Source

Description: This command sets or queries the instrument's remote firmware update package list source. This source must be an http URL that is accessible to the instrument over the current network interface.

The set version of this command takes a string that spells the URL to the package list file. The query version returns that string. The default value points to the repository on anritsu.com where official firmware packages for this instrument are distributed. Most users will never need to change this setting from the default.

For example, the url value either set or query returned may be:

http://softwareupdates.anritsu.com/en-us/ms209xa.txt

The package list file must be a JSON file that has a minimum structure. The root object must contain a "version" string (which is reserved but currently unused), a "packages" array of zero or more package objects (each at minimum have a "name", "url", "version", and "model"), and a "default" string that matches one of the package names, or is "". The version and model attributes have semantic meaning, but the name and filename (url) can be anything. An example package list file is shown below:


{

"default": "V2021.11.2 (MS209xA)",

"packages": [

{

"name": "V2021.11.2 (MS209xA)",

"version": "V2021.11.2",

"model": "MS209xA",

"url": "http://softwareupdates.anritsu.com/en-us/MS2090A_V2021.11.2.raucb"

}

],

"version": "1.0.0"

}


or, as with MS2710xA:


{

"default": "V1.0.0 (MS2710xA)",

"packages": [

{

"name": "V1.0.0 (MS2710xA)",

"version": "V1.0.0",

"model": "MS2710xA",

"url": "http://softwareupdates.anritsu.com/en-us/MS2710xA_V1.0.0.tar"

}

],

"version": "1.0.0"

}

When the package source is set, the instrument will attempt to download the file and validate it according to the above rules before making the change permanent. If the file cannot be accessed or the file does not conform to the minimum criteria, a device specific error will be added to the error queue and the package source will remain unchanged.

Parameter(s): <string>

**:SYSTem:FIRMware:VERSion?**

Title: Firmware Version

Description: This command queries the device firmware version.

**:SYSTem:GPS:SEARch:MODE <GGLonass|GGALileo|GBEidou|GGGalileo>**
**:SYSTem:GPS:SEARch:MODE?**

Title: GPS Satellite System Search Mode

Description: Sets and queries the GPS Satellite system search option.

The satellite system search options can be:

GPS: GPS only

GGLonass: GPS + GLONASS

GGALileo: GPS + GALILEO

GBEidou: GPS + BEIDOU

GGGalileo: GPS + GLONASS + GALILEO

Parameter(s): <GGLonass | GGALileo | GBEidou | GGGalileo>

Query Return: GGL | GGAL | GBE | GGG

Default Value: GGGalileo

**:SYSTem:GPS:STATe <0 | 1 | ON | OFF>**
**:SYSTem:GPS:STATe?**

Title: GPS State

Description: Enable or disable the GPS. When the GPS is OFF, the GPS module is placed into standby to save power, and features which depend on the GPS may be unusable.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:SYSTem:GPS:VOLTage <numeric_value>**
**:SYSTem:GPS:VOLTage?**

Title: GPS Voltage

Description: Supplies either 3.3V or 5V to power the attached GPS

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 3.3

Range: 3.3 to 5

**:SYSTem:LOG:ERRor?**

Title: System Error Log

Description: This command retrieves the error log. The log is encrypted and can be saved to a file and sent to Anritsu Service if required.

**:SYSTem:MACaddress?**

Title: MAC Address

Description: Query the MAC address of the instrument.

Setting the MAC address is only available to customer service.

**:SYSTem:OPTions?**

Title: Query Options

Description: This command retrieves the options that are currently set. The response is returned in "/" delimited form.

**:SYSTem:OPTions:ALL?**

Title: Query All Options

Description: This command returns all installed options with their metadata in JSON format. The expiration date time is in UTC time and in a string format of the form yyyy-MM-ddTHH:mm:ss corresponding to the ISO 8601 specification.

Example:

[{"expirationDate":"","option":"0090", "state":"active", "description":"Gated Sweep"},

{"expirationDate":"2022-07-05T19:00:42Z","option":"9103", "state":"pendingReboot", "description":"50 MHz Analysis Bandwidth" },

{"expirationDate":"","option":"9431", "state":"available", "description":"Coverage Mapping"}]

**:SYSTem:OPTions:CONFig?**

Title: Option Configuration

Description: This command returns a quoted string of characters readable only by Anritsu Customer Service. Only instrument configuration information is returned. No setup information is included.

**:SYSTem:OPTions:TIMed:ACTivate <string>**

Title: Activate Timed Options

Description: This command activates an existing Timed Option.

Parameter(s): <string>

**:SYSTem:OPTions:UPDate:REMote**

Title: Option Update Remote

Description: This command retrieves all available Licenses for given instrument and installs them.

### :SYSTem:OPTions:UPGRade <string>

Title: Option Upgrade

Description: This command upgrades the options of the device with a valid license key provided as a quoted string parameter. The device must be rebooted immediately after this command for the installation of the new options to complete. The device must not be powered off while the command is executing. When the operation completes, a success message will be shown and request to reboot the device. The recommended way to restart the device safely is by sending the *RST command after this command. This command might report the following error code: -200: The command failed to complete. A detailed error message will be provided in the response of :SYStem:ERRor[:NEXT]? command. -2: The command failed because the options provided in the option file are incompatible with this instrument.

Parameter(s): <string>

### :SYSTem:PASSword:CDISable <string>

Title: Enable Remote Password

Description: Enable the remote password state.

All existing remote connections to the instrument on a raw socket will be closed.

Future remote connections to the instrument must be on a secure socket via port 9003 and must provide a matching password (using :SYSTem:PASSword:CENable command) in order to gain access to other SCPI commands on the instrument.

Parameter:

1. StringValue: The remote password encrypted using OpenSSL RSA encryption algorithm with Anritsu Remote Password public key and OAEP padding mode in a base64 string. Use :SYSTem:PASSword:SSLKey to get Anritsu Remote Password public key.

Sending an invalid password generates error -221 (setting conflict).

The command is only available if the Secure Communication option is installed.

Parameter(s): <string>

### :SYSTem:PASSword:CENable <string>

Title: Enable Password Protected Connection

Description: Unlocks a particular connection to allow all SCPI commands to be sent via that connection when remote password state is ON.

Parameter:

1. StringValue: The remote password encrypted using OpenSSL RSA encryption algorithm with Anritsu Remote Password public key and OAEP padding mode in a base64 string. Use :SYSTem:PASSword:SSLKey to get Anritsu Remote Password public key.

Sending an invalid password generates error -221 (setting conflict).

Send an empty string as the password parameter if the remote password has not been set but the remote password state is ON.

The command is only available if the Secure Communication option is installed.

Parameter(s): <string>

### :SYSTem:PASSword:CENable:ALL <string>

Title: Disable Remote Password

Description: Disable the remote password state.

Remote connections to the instrument on a secure socket via port 9003 will be disabled.

Remote connections to the instrument on a raw socket via port 9001 are enabled.

Parameter:

1. StringValue: The remote password encrypted using OpenSSL RSA encryption algorithm with Anritsu Remote Password public key and OAEP padding mode in a base64 string. Use :SYSTem:PASSword:SSLKey to get Anritsu Remote Password public key.

Sending an invalid password generates error -221 (setting conflict).

The command is only available if the Secure Communication option is installed.

Parameter(s): <string>

### :SYSTem:PASSword:CONNection:STATe?

Title: Remote Password Connection State

Description: Returns whether the particular connection the command is being sent from is locked behind a password.

The query will return 0 if the connection is still locked behind a password or 1 if SCPI commands can be sent.

To unlock the connection for all SCPI commands, use :SYSTem:PASSword:CENable.

The command is only available if the Secure Communication option is installed.

### :SYSTem:PASSword:INTegrity?

Title: Remote Password Integrity

Description: Returns whether the remote password checksum is good or bad. The query command will return one of the two following values:

Good: The password integrity is good and can be used as normal.

Bad: The password has failed checksum and will possibly no longer match properly.

Commands that require the entry of the password (:SYSTem:PASSword:CENable, :SYSTem:PASSword:CENable:ALL, :SYSTem:PASSword:CDISable, and :SYSTem:PASSword:NEW)

may no longer function.

The remote password can be reset in one of two ways:

1. On instruments with a local GUI, a new password can be set WITHOUT triggering a master reset.

2. To reset the password remotely, the system override reset_remote_password command will have to be used.

The command can be sent through a connection to the 8001(normal socket)/8002(secure socket) port of the instrument

and has the format:

reset_remote_password,<system_override_specific_password>

Please read the user manual for more details about the system override password.

Note that resetting the password through system override will clear the password, set the remote password state to OFF,

and trigger a master reset (which clears all user files).

### :SYSTem:PASSword:NEW <string>,<string>

Title: Remote Password Set

Description: Set a new remote password.

Paramters:

1. StringValue: The current encrypted password in base64 string.

2. StringValue: The new encrypted password in base64 string.

The current and new passwords must be encrypted using OpenSSL RSA encryption algorithm with Anritsu Remote Password public key and OAEP padding mode. Use :SYSTem:PASSword:SSLKey to get Anritsu Remote Password public key.

Sending an invalid current password generates error -221 (setting conflict).

Sending a new password not meeting the password complexity rules generates error -232 (invalid format).

The command is only available if the Secure Communication option is installed.

On a secure socket connection, the command is only accessible after command :SYSTem:PASSword:CENable has been received with a valid password.

The remote password is empty by default.

A new password must pass the following checks to be accepted:

1. It must have 12 characters minimum and 64 characters maximum.

2. It must have letters and number(s).

3. It must have both lowercase and uppercase letters.

4. It must have at least one special character. Valid special characters are @#$_&-+=%*"':;!?() ,.`~/{}[]<>

5. It must not have 5 or more increasing sequential numbers (such as 12345).

6. It must not have 5 or more repeated characters (such as aaaaa).

Parameter(s): <string>,<string>

### :SYSTem:PASSword:SET?

Title: Remote Password is Set

Description: Returns whether the remote password is set.

The command will return 1 if the remote password is set to a non-empty value and 0 if the password is empty.

### :SYSTem:PASSword:SSLKey?

Title: Remote Password Public Key

Description: Return Anritsu OpenSSL RSA public key (in PEM format). The key is used to encrypt the remote password parameter for password protected commands such as :SYSTem:PASSword:CENable.

The key is returned in IEEE definite length arbitrary block format:

<#AX><key> where A is a single ascii digit (1-9) describing the number of bytes in the length section (X).

X is ascii string of digits (0-9) describing the length of the <key>.

The command is only available if the Secure Communication option is installed.

**:SYSTem:PASSword:STATe?**

Title: Remote Password State

Description: Returns whether the instrument is locked behind a remote password. While remote password is turned on, only

commands that are not password protected will be processed until either the connection itself is unlocked using

:SYSTem:PASSword:CENable or if the remote password state is turned off using :SYSTem:PASSword:CENable:ALL

Query Return: 0 | 1

Default Value: OFF

**:SYSTem:PRESet**

Title: Preset

Description: This command presets parameters in the current application to their factory default values.

**:SYSTem:PRESet:MODE**

Title: Preset Mode

Description: This command presets parameters in the current mode to their factory default values.

**:SYSTem:REBoot**

Title: System Reboot

Description: This command saves the current state of the instrument and reboots the instrument.

**:SYSTem:REMote:DISPlay:LOCK <0 | 1 | ON | OFF>**
**:SYSTem:REMote:DISPlay:LOCK?**

Title: Remote Control State

Description: This command restricts front panel activity on the instrument until remote control is turned off either through this SCPI command or through a button on the front panel.

The query will return 1 if the pop-up is active, or 0 if the pop-up is inactive.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:SYSTem:SETup:SHORtcuts <numeric_value>,<string>,[<string>]**
**:SYSTem:SETup:SHORtcuts?**

Title: Set System Footer Shortcuts

Description: This command allows saving up to 5 setup files into shortcuts will display at the base of the user interface.

This will allows easy access to loading frequently used setups. The shortcuts are numbered 1-5 from left to right.

Parameters:

1. Numeric Value: The shortcut number that you would like to set

2. Name value: The name that will be displayed on the user interface screen.

3. File path: The file path (including the file name) relative to the MSUS. Without Secure Data option, MSUS is the Internal device. With Secure Data option installed, MSUS is a connected USB device. File extension can be specified, but is not required. Only .stp files will be accepted. If left blank, it will save the current setup and the file will be saved to a default file

Example usage:

Set shortcut #3 to file example_file.stp in your instrument file folder: SYST:SET:SHOR 1,"EMF Setup 1", "example_file.stp"

Set the current setup of Spectrum to shortcut #1: SYST:SET:SHOR 2,"SPEC 10 MHz"

SYST:SET:SHORT? will return as:
[{"app":"SPA","file":"example_file.stp.stp","mode":"SPEC","name":"EMF Setup 1"},{"app":"SPA","file":"SPEC_10_MHz.stp","mode":"SPEC","name":"SPEC 10 MHz"},{"app":"N/A","file": "N/A","mode":"N/A","name":"SHORTCUT"}]

This command is disabled if Secure Data Option is available and enabled.

Parameter(s): <numeric_value>,<string>,[<string>]

Default Value:

[{"app":"N/A","file":"N/A","mode":"N/A","name":"SHORTCUT","status":false,"lastLoaded":false}]

**:SYSTem:SETup:SHORtcuts:CLEar**

Title: Load System Footer Shortcuts

Description: This command will clear all of the System Setup shortcuts returned by SYST:SET:SHOR?

This command is disabled if Secure Data Option is available and enabled.

**:SYSTem:SETup:SHORtcuts:DEL <numeric_value>**

Title: Delete a Single System Footer Shortcut

Description: This command will delete a single System Setup shortcuts returned by SYST:SET:SHOR? at the given index.

This command is disabled if Secure Data Option is available and enabled.

Parameter(s): <numeric_value>

**:SYSTem:SETup:SHORtcuts:LOAD <numeric_value>**

Title: Load System Footer Shortcuts

Description: This command will load the setup saved at the given shortcut number.

Parameters:

1. Numeric Value: The shortcut number that you would like to load

Usage: SYST:SET:SHOR:LOAD 2

This command is disabled if Secure Data Option is available and enabled.

Parameter(s): <numeric_value>

**:SYSTem:SETup:SHORtcuts:REName <numeric_value>,<string>**

Title: Rename System Footer Shortcuts

Description: This command allows renaming a shortcut display name.

Parameters:

1. Numeric Value: The shortcut number that you would like to set

2. Name value: The new name that will be displayed on the user interface screen for that given shortcut number.

Usage: SYST:SET:SHOR:LOAD 2, "Example Name"

SYST:SET:SHORT? will return as:
[{"app":"SPA","file":"example_file.stp.stp","mode":"SPEC","name":"EMF Setup 1"},{"app":"SPA","file":"SPEC_10_MHz.stp","mode":"SPEC","name":"Example Name"},{"app":"N/A","file": "N/A","mode":"N/A","name":"SHORTCUT"}]

This command is disabled if Secure Data Option is available and enabled.

Parameter(s): <numeric_value>,<string>

**:SYSTem:TEMPerature?**

Title: Get CPU Temperature

Description: This command returns the CPU temperature in Celsius.

**:SYSTem:TIME <numeric_value>,<numeric_value>,<numeric_value>**
**:SYSTem:TIME?**

Title: System Time

Description: This command sets and queries the system's internal clock. Changing the system time does not affect the system time zone (if time zone is available). The three parameters for this command are <hour>,<minute> and <second>. The query response message shall consist of three fields separated by commas: <hour>,<minute>,<second>. Note: Some products only support the query version of this command.

Parameter(s): <numeric_value>,<numeric_value>,<numeric_value>

**:SYSTem:TIME:SYNC <AUTO|MANual>**
**:SYSTem:TIME:SYNC?**

Title: Sync System Time

Description: This command sets the system to either automatically synchronize with GPS/Internet server or manually adjust the time setting. Valid parameter values are AUTO|MANual. When AUTO is selected, GPS takes priority if it is present and turned on (see command :SYSTem:GPS:STATe to set GPS on or off). When the time setting is AUTO synchronized, the current time zone setting is not changed.

Parameter(s): <AUTO|MANual>

Query Return: AUTO|MAN

Default Value: AUTO

**:SYSTem:TIME:SYNC:STATus?**

Title: Sync Time Status

Description: This command query the last system time synchronization status. The return value range are INT, GPS, NONE, or NA.

INT - the system time was successfully synchronized with internet server.

GPS - the system time was successfully synchronized with GPS device time.

NONE - the system time was set to automatically synchronize with GPS/internet. However, it was unable to synchronize with the time source. It may indicates the time source connection is not available or time source is not ready such as GPS is off.

NA - the system time was set to manually update, thus synchronization status is not available.

Query Return: INT|GPS|NONE|NA

Default Value: INTernet

**:SYSTem:TIME:ZONE <string>**
**:SYSTem:TIME:ZONE?**

Title: System Time Zone

Description: This command sets and queries the system time zone. The time zone parameter is in string format of "(UTC+/-offset) Area/Location" as returned from the :SYSTem:TIME:ZONE:CATalog command. Changing the system time zone affects the system date/time.

Parameter(s): <string>

Default Value: (UTC-08:00) America/Los_Angeles

**:SYSTem:TIME:ZONE:CATalog?**

Title: System Time Zone Catalog

Description: This command queries the available system time zone catalog. The query returns comma delimited description for time zone sorted by UTC offset. Each item is in the format of "(UTC+/-offset) Area/Location" where Area/Location is defined in the tz database

**:SYSTem:UNIT:NAMe <string>**
**:SYSTem:UNIT:NAMe?**

Title: System Unit Name

Description: This command sets/queries the unit name of the current system.The name of the unit must not be greater than 45 ASCII characters

Parameter(s): <string>

**:TEST:SELFtest?**

Title: Self Test

Description: Perform a self-test and return the results. The response is formatted as a JSON (http://json.org/) array of name,value pair result objects. Some result objects also include a 'status' property that will be either 'pass' or 'fail' based on whether the criteria for that test was met.

For tests of voltages, the test passes if the measured voltage is within 10% of the expected voltage.

**:TRIGger[:SEQuence]:PORT<n>[:DIRection] <IN|OUT>**
**:TRIGger[:SEQuence]:PORT<n>[:DIRection]?**

Title: Trigger Port Direction

Description: This command configures the direction of the trigger ports, which can behave as either trigger in or trigger out. The command suffix selects which of the two trigger ports to configure. Note that port 1 doubles as a reference out port and cannot be a reference out and trigger port at the same time. To use port one as a trigger port the reference output state must be turned off (see the 10MHz Reference Output State) command. For port 2 when set to trigger out, the GPS PPS will be the pulse output. Command not available on the MS2080A or MS2070A, where the direction is always IN.

Parameter(s): <IN|OUT>

PORT Suffix Range: 1-2, Default = 1

## 2-3 Spectrum Analyzer Commands

**:ABORt**

Title: Abort

Description: Resets the trigger system. This has the effect of aborting the sweep or any measurement that is currently in progress.

Additionally, any pending operation flags that were set by initiation of the trigger system will be set to false.

If :INITiate:CONTinuous is OFF (i.e. the instrument is in single sweep mode), send the command :INITiate[:IMMediate] to trigger the next sweep.

If :INITiate:CONTinuous is ON (i.e. the instrument is in continuous sweep mode) a new sweep will start immediately.

**:CALCulate:LIMit<n>:COMMent <string>**
**:CALCulate:LIMit<n>:COMMent?**

Title: Limit Comment

Description: Associates a user-defined comment with each limit. The set version of this command takes a single parameter that is a string containing the desired comment. The query version returns the comment that is set.

If a limit of the specified number does not already exist, a default limit will be created first (having empty

data, with state set to ON).

Note that this command is not available for Spectral Emission Mask measurement.

Parameter(s): <string>

LIMit Suffix Range: 1-10, Default = 1

**:CALCulate:LIMit<n>:CONTrol[:DATA] <numeric_value> {HZ | KHZ | MHZ | GHZ}, {<numeric_value> {HZ | KHZ | MHZ | GHZ}}, ...**
**:CALCulate:LIMit<n>:CONTrol[:DATA]?**

Title: Limit Control Data

Description: This command sets or queries the control data (X-axis values) for the specified limit.

The numeric suffix on LIMit specifies which limit number to query or set data. If the suffix is omitted, the command will refer to limit 1.

If a limit of the specified number does not already exist, a default limit will be created first (having empty data, with state set to ON).

This command takes a list of one or more frequency values as parameters. For example, CALC:LIM1:CONT:DATA 1 MHz, 2MHz, 3MHz

Note that it is permitted to set one or more invalid 'placeholder' values of Not-A-Number (NAN). The placeholder value for NAN is 9.91e37. If a control data point contains a value of 9.91e37, limit line interpolation from the previous data point, and to the next data point, will not occur. This is useful for defining discontiguous (or segmented) limit lines within a single limit.

For example, to define a limit line of two discontiguous segments, one from 1 MHz to 10 MHz, and another from 20 MHz to 30 MHz, send the following control data: CALC:LIM1:CONT:DATA 1MHz, 10MHz, 9.91e37, 20MHz, 30MHz

Note that the upper (or lower) data, if used, should contain the same amount of points as the control data (see CALC:LIM:FAIL? for details on what happens when this is not true). Thus, when using placeholders in control data, it is recommended that placeholders are also used in the upper (or lower) data. For example, CALC:LIM1:CONT:UPP 0dBm, 10dBm, 9.91e37, 10dBm, 0dBm

The exact value for the middle point does not matter, as interpolation will always be skipped due to the control data containing NAN, but using NAN for the corresponding upper (or lower) data value will make it easier to remember that the point is a placeholder.

Note that this command is not available for Spectral Emission Mask measurement.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}, {<numeric_value> {HZ | KHZ | MHZ | GHZ}}, ...

LIMit Suffix Range: 1-10, Default = 1

**:CALCulate:LIMit<n>:CONTrol:MODE <ABSolute|RELative>**
**:CALCulate:LIMit<n>:CONTrol:MODE?**

Title: Limit Control Mode

Description: Choose either ABSolute or RELative limit.

In ABSolute mode, the control value of the limit line is defined by absolute physical values (Hz).

In RELative mode, the control value of the limit line is relative to the center frequency (Hz).

If a limit of the specified number does not already exist, a default limit will be created first (having empty

data, with state set to ON).

Note that this command is not available for Spectral Emission Mask measurement.

Parameter(s): <ABSolute | RELative>

Query Return: ABS | REL

LIMit Suffix Range: 1-10, Default = 1

Default Value: ABSolute

**:CALCulate:LIMit<n>:CONTrol:SHIFt <numeric_value> {HZ | KHZ | MHZ | GHZ}**

Title: Limit Control Shift

Description: Move a limit line along the control axis. This command changes the value of :CALCulate<n>:LIMit<k>:CONTrol[:DATA].

Issuing this command multiple times will change the limits each time. For example, sending

CALC:LIM1:CONT:SHIFT 1 Hz

CALC:LIM1:CONT:SHIFT 1 Hz

CALC:LIM1:CONT:SHIFT 1 Hz

CALC:LIM1:CONT:SHIFT 1 Hz

CALC:LIM1:CONT:SHIFT 1 Hz

will shift the control axis by 5 Hz

If a limit of the specified number does not already exist, a default limit will be created first (having empty data, with state set to ON).

Note that this command is not available for Spectral Emission Mask measurement.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

LIMit Suffix Range: 1-10, Default = 1

### :CALCulate:LIMit<n>:COPY <numeric_value>

Title: Limit Copy

Description: Copies a limit line eg: CALC:LIM1:COPY 2 Copies limit 1 to line 2

If a limit of the specified number does not already exist, a default limit will be created first (having empty data, with state set to ON).

Note that this command is not available for Spectral Emission Mask measurement.

Parameter(s): <numeric_value>

LIMit Suffix Range: 1-10, Default = 1

### :CALCulate:LIMit<n>:DELete

Title: Limit Delete

Description: Deletes a limit line

If a limit of the specified number does not already exist, a default limit will be created first (having empty data, with state set to ON).

Note that this command is not available for Spectral Emission Mask measurement.

LIMit Suffix Range: 1-10, Default = 1

### :CALCulate:LIMit<n>:FAIL?

Title: Limit Fail

Description: This command queries the result of a limit check.

All traces that have had checking enabled (via CALC:TRAC:CHEC) will be evaluated against the upper and lower data of the specified limit, unless the limit STATe is OFF (in which case this command will always return 0), or the upper or lower STATe is OFF (in which case only the data with STATe ON will be checked).

If the sweep has not completed yet, the partial trace will be evaluated. If INITiate:CONTinuous is ON, a snapshot of the trace at the time this command was received will be evaluated against the limit. This command returns 1 if any of the checked traces violate the limit, otherwise it returns 0.

When a limit is evaluated, there are some rules that are followed if the cardinality of the limits control, upper, and lower data are not equal. If there are fewer control data points than upper or lower data points, then only the first n upper or lower points will be evaluated, where n is the number of control points. If there are fewer upper or lower points than control points, then the effective number of upper or lower points will be increased to the number of control points, with the 'extrapolated' points having a value equal to the last upper or lower point.

If a limit of the specified number does not already exist, a default limit will be created first (having empty data, with state set to ON).

LIMit Suffix Range: 1-10, Default = 1

**:CALCulate:LIMit<n>:LOWer[:DATA] <numeric_value> {<amplitude_units>}, {<numeric_value> {<amplitude_units>}}, ...**
**:CALCulate:LIMit<n>:LOWer[:DATA]?**

Title: Lower Limit Data

Description: This command sets or queries the lower limit data (Y-axis values) for the specified limit.

The numeric suffix on LIMit specifies which limit number to query or set data. If the suffix is omitted, the command will refer to limit 1.

If a limit of the specified number does not already exist, a default limit will be created first (having empty data, with state set to ON).

This command takes a list of one or more amplitude values as parameters. For example,

CALC:LIM1:LOW:DATA 1 dBm, 2dBm, 3 dBm

Note that it is permitted to set one or more data values of +/- infinity. The placeholder value for +/- infinity is +/-9.9e37. If a lower data point contains a value of +/-9.9e37, the amplitude at that point will be treated as if it were +/-infinity (i.e. the lower limit will either always fail or always pass at that point).

It is also permitted to set one or more invalid 'placeholder' values of Not-A-Number (NAN). This is useful for defining discontiguous (or segmented) limit lines within a single limit.

For details, and an example, of using placeholder values, see CALC:LIM:CONT:DATA.

Note that if the units for the limit values are specified (as shown in the example above), they must have the same unit.

Note that this command is not available for Spectral Emission Mask measurement.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {<amplitude_units>}, {<numeric_value> {<amplitude_units>}}, ...

LIMit Suffix Range: 1-10, Default = 1

**:CALCulate:LIMit<n>:LOWer:MODE <ABSolute|RELative>**
**:CALCulate:LIMit<n>:LOWer:MODE?**

Title: Limit Lower Mode

Description: Choose either ABSolute or RELative limit.

In ABSolute mode, the control value of the limit line is defined by absolute physical values (dBm).

In RELative mode, the control value of the limit line is relative to the reference level (dB).

If a limit of the specified number does not already exist, a default limit will be created first (having empty data, with state set to ON).

Note that this command is not available for Spectral Emission Mask measurement.

Parameter(s): <ABSolute|RELative>

Query Return: ABS|REL

LIMit Suffix Range: 1-10, Default = 1

Default Value: ABSolute

**:CALCulate:LIMit<n>:LOWer:SHIFt <numeric_value> {DB}**

Title: Lower Limit Shift

Description: Move the lower limit up or down by a relative amplitude. This command changes the value of :CALCulate<n>:LIMit<k>:LOWer[:DATA].

Issuing this command multiple times will change the limits each time. For example, sending

CALC:LIM1:LOW:SHIFT 1 dB

CALC:LIM1:LOW:SHIFT 1 dB

CALC:LIM1:LOW:SHIFT 1 dB

CALC:LIM1:LOW:SHIFT 1 dB

CALC:LIM1:LOW:SHIFT 1 dB

will shift the lower limit by 5 dB

If a limit of the specified number does not already exist, a default limit will be created first (having empty data, with state set to ON).

Note that this command is not available for Spectral Emission Mask measurement.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {DB}

LIMit Suffix Range: 1-10, Default = 1

**:CALCulate:LIMit<n>:LOWer:STATe <0 | 1 | ON | OFF>**
**:CALCulate:LIMit<n>:LOWer:STATe?**

Title: Lower Limit State

Description: Turns ON or OFF the lower limit.

If a limit of the specified number does not already exist, a default limit will be created first (having empty data, with state set to ON).

Note that this command is not available for Spectral Emission Mask measurement.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

LIMit Suffix Range: 1-10, Default = 1

Default Value: OFF

**:CALCulate:LIMit<n>:NAME <string>**
**:CALCulate:LIMit<n>:NAME?**

Title: Limit Name

Description: Associates a user-defined name with each limit.

The set version of this command takes a single parameter that is a string containing the desired limit name. The query version returns the name that is set.

If a limit of the specified number does not already exist, a default limit will be created first (having empty data, with state set to ON).

Note that this command is not available for Spectral Emission Mask measurement.

Parameter(s): <string>

LIMit Suffix Range: 1-10, Default = 1

**:CALCulate:LIMit<n>:STATe <0 | 1 | ON | OFF>**
**:CALCulate:LIMit<n>:STATe?**

Title: Limit State

Description: Turns the limit check for a specific limit ON or OFF.

If a limit of the specified number does not already exist, a default limit will be created first (having empty data, with state set to ON).

Note that this command is not available for Spectral Emission Mask measurement.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

LIMit Suffix Range: 1-10, Default = 1

Default Value: OFF

**:CALCulate:LIMit<n>:TRACe<n>:CHECk <0 | 1 | ON | OFF>**
**:CALCulate:LIMit<n>:TRACe<n>:CHECk?**

Title: Limit Trace Check

Description: This command turns the limit check for a specific trace on and off.

If a limit of the specified number does not already exist, a default limit will be created first (having empty data, with state set to ON).

Note that this command is not available for Spectral Emission Mask measurement.

Parameter(s): <0 | 1 | ON | OFF>

LIMit Suffix Range: 1-10, Default = 1

TRACe Suffix Range: 1-6, Default = 1

**:CALCulate:LIMit<n>:UPPer[:DATA] <numeric_value> {<amplitude_units>}, {<numeric_value> {<amplitude_units>}}, ...**
**:CALCulate:LIMit<n>:UPPer[:DATA]?**

Title: Upper Limit Data

Description: This command sets or queries the upper limit data (Y-axis values) for the specified limit.

The numeric suffix on LIMit specifies which limit number to query or set data. If the suffix is omitted, the command will refer to limit 1.

If a limit of the specified number does not already exist, a default limit will be created first (having empty data, with state set to ON).

This command takes a list of one or more amplitude values as parameters. For example, CALC:LIM1:UPP:DATA 1 dBm, 2dBm, 3 dBm

Note that it is permitted to set one or more data values of +/- infinity. The placeholder value for +/- infinity is +/-9.9e37. If an upper data point contains a value of +/-9.9e37, the amplitude at that point will be treated as if it were +/-infinity (i.e. the upper limit will either always pass or always fail at that point).

It is also permitted to set one or more invalid 'placeholder' values of Not-A-Number (NAN). This is useful for defining discontiguous (or segmented) limit lines within a single limit.

For details, and an example, of using placeholder values, see CALC:LIM:CONT:DATA.

Note that if the units for the limit values are specified (as shown in the example above), they must have the same unit.

Note that this command is not available for Spectral Emission Mask measurement.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {<amplitude_units>}, {<numeric_value> {<amplitude_units>}}, ...

LIMit Suffix Range: 1-10, Default = 1

**:CALCulate:LIMit<n>:UPPer:MODE <ABSolute|RELative>**
**:CALCulate:LIMit<n>:UPPer:MODE?**

Title: Upper Limit Mode

Description: Choose either ABSolute or RELative limit.

In ABSolute mode, the control value of the limit line is defined by absolute physical values (dBm).

In RELative mode, the control value of the limit line is relative to the reference level (dB).

If a limit of the specified number does not already exist, a default limit will be created first (having empty

data, with state set to ON).

Note that this command is not available for Spectral Emission Mask measurement.

Parameter(s): <ABSolute|RELative>

Query Return: ABS|REL

LIMit Suffix Range: 1-10, Default = 1

Default Value: ABSolute

**:CALCulate:LIMit<n>:UPPer:SHIFt <numeric_value> {DB}**

Title: Upper Limit Shift

Description: Move the upper limit up or down by a relative amplitude. This command changes the value of :CALCulate<n>:LIMit<k>:UPPer[:DATA].

Issuing this command multiple times will change the limits each time. For example, sending

CALC:LIM1:UPP:SHIFT 1 dB

CALC:LIM1:UPP:SHIFT 1 dB

CALC:LIM1:UPP:SHIFT 1 dB

CALC:LIM1:UPP:SHIFT 1 dB

CALC:LIM1:UPP:SHIFT 1 dB

will shift the upper limit by 5 dB

If a limit of the specified number does not already exist, a default limit will be created first (having empty data, with state set to ON).

Note that this command is not available for Spectral Emission Mask measurement.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {DB}

LIMit Suffix Range: 1-10, Default = 1

**:CALCulate:LIMit<n>:UPPer:STATe <0 | 1 | ON | OFF>**
**:CALCulate:LIMit<n>:UPPer:STATe?**

Title: Upper Limit State

Description: Turns ON or OFF the upper limit.

If a limit of the specified number does not already exist, a default limit will be created first (having empty data, with state set to ON).

Note that this command is not available for Spectral Emission Mask measurement.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

LIMit Suffix Range: 1-10, Default = 1

Default Value: OFF

**:CALCulate:LIMit:ACTive?**

Title: Limit Active

Description: Queries the numbers of all active limit lines in ascending order. This command returns a empty string if no limits are active.

Note that this command is not available for Spectral Emission Mask measurement.

**:CALCulate:ACPower:LIMit:ADJacent:ABSolute <numeric_value> {<amplitude_units>}**
**:CALCulate:ACPower:LIMit:ADJacent:ABSolute?**

Title: Adjacent Channel Power Adjacent Absolute Limit
Description: Sets and queries the amplitude of absolute limit for ACPR adjacent power.
See UNIT:POWer to query or set the current amplitude units.
Parameter(s): <numeric_value> {<amplitude_units>}
Query Return: Numeric (<amplitude_units>)
Default Value: 10 dBm
Default Unit: dBm
Range: -200 dBm to 200 dBm

**:CALCulate:ACPower:LIMit:ADJacent:LOWer:FAIL?**

Title: ACPR Limit Adjacent Lower Fail
Description: This command queries the result of a limit check
on lower adjacent channel power

**:CALCulate:ACPower:LIMit:ADJacent:RELative <numeric_value> {DB}**
**:CALCulate:ACPower:LIMit:ADJacent:RELative?**

Title: Adjacent Channel Power Adjacent Relative Limit
Description: Sets and queries the amplitude of relative limit for ACPR adjacent power.
See UNIT:POWer to query or set the current amplitude units.
Parameter(s): <numeric_value> {DB}
Query Return: Numeric (dB)
Default Value: 10 dB
Default Unit: dB
Range: -200 dB to 200 dB

**:CALCulate:ACPower:LIMit:ADJacent:UPPer:FAIL?**

Title: ACPR Limit Adjacent Upper Fail
Description: This command queries the result of a limit check
on upper adjacent channel power

**:CALCulate:ACPower:LIMit:ALTernate:ABSolute <numeric_value> {<amplitude_units>}**
**:CALCulate:ACPower:LIMit:ALTernate:ABSolute?**

Title: Adjacent Channel Power Alternate Absolute Limit
Description: Sets and queries the amplitude of absolute limit for ACPR alternate power.
See UNIT:POWer to query or set the current amplitude units.
Parameter(s): <numeric_value> {<amplitude_units>}
Query Return: Numeric (<amplitude_units>)
Default Value: 10 dBm
Default Unit: dBm
Range: -200 dBm to 200 dBm

**:CALCulate:ACPower:LIMit:ALTernate:LOWer:FAIL?**

Title: ACPR Limit Alternate Lower Fail
Description: This command queries the result of a limit check
on lower alternate channel power

**:CALCulate:ACPower:LIMit:ALTernate:RELative <numeric_value> {DB}**
**:CALCulate:ACPower:LIMit:ALTernate:RELative?**

Title: Adjacent Channel Power Alternate Relative Limit
Description: Sets and queries the amplitude of relative limit for ACPR alternate power.
See UNIT:POWer to query or set the current amplitude units.
Parameter(s): <numeric_value> {DB}
Query Return: Numeric (dB)
Default Value: 10 dB
Default Unit: dB
Range: -200 dB to 200 dB

**:CALCulate:ACPower:LIMit:ALTernate:UPPer:FAIL?**

Title: ACPR Limit Alternate Upper Fail
Description: This command queries the result of a limit check
on upper alternate channel power

**:CALCulate:ACPower:LIMit:FAIL?**

Title: ACPR Limit Fail
Description: This command queries the result of a limit check
on adjacent channel power ratio

**:CALCulate:ACPower:LIMit:MAIN <numeric_value> {<amplitude_units>}**
**:CALCulate:ACPower:LIMit:MAIN?**

Title: Adjacent Channel Power Main Limit
Description: Sets and queries the amplitude of limit for main power.
See UNIT:POWer to query or set the current amplitude units.
Parameter(s): <numeric_value> {<amplitude_units>}
Query Return: Numeric (<amplitude_units>)
Default Value: 10 dBm
Default Unit: dBm
Range: -200 dBm to 200 dBm

**:CALCulate:ACPower:LIMit:MODE <ABSolute|RELative>**
**:CALCulate:ACPower:LIMit:MODE?**

Title: Adjacent Channel Power Limit Mode
Description: Sets the ACPR limit mode to be absolute or relative
Parameter(s): <ABSolute|RELative>
Query Return: ABS|REL
Default Value: ABSolute

**:CALCulate:ACPower:LIMit:STATe <0 | 1 | ON | OFF>**
**:CALCulate:ACPower:LIMit:STATe?**

Title: Adjacent Channel Power Limit State
Description: Sets the ACPR limit state to be ON or OFF
Parameter(s): <0 | 1 | ON | OFF>
Query Return: 0 | 1
Default Value: OFF

**:CALCulate:CHPower:LIMit <numeric_value> {<amplitude_units>}**
**:CALCulate:CHPower:LIMit?**

Title: Channel Power Limit
Description: Sets and queries the amplitude of limit on channel power.
See UNIT:POWer to query or set the current amplitude units.
Parameter(s): <numeric_value> {<amplitude_units>}
Query Return: Numeric (<amplitude_units>)
Default Value: 10 dBm
Default Unit: dBm
Range: -200 dBm to 70 dBm

**:CALCulate:CHPower:LIMit:FAIL?**

Title: Channel Power Limit Fail
Description: This command queries the result of a limit check
on channel power

**:CALCulate:CHPower:LIMit:PSDensity <numeric_value> {<amplitude_units>}**
**:CALCulate:CHPower:LIMit:PSDensity?**

Title: Channel Power Spectral Density Limit

Description: Sets and queries the amplitude of limit on channel power spectral density.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {<amplitude_units>}

Query Return: Numeric (<amplitude_units>)

Default Value: 10 dBm

Default Unit: dBm

Range: -200 dBm to 70 dBm

**:CALCulate:CHPower:LIMit:PSDensity:STATe <0 | 1 | ON | OFF>**
**:CALCulate:CHPower:LIMit:PSDensity:STATe?**

Title: Channel Power Spectral Density Limit State

Description: Sets and queries the state of limit on channel power spectral density. The set form of this command sets the limit state to be ON or OFF

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:CALCulate:CHPower:LIMit:PSD:FAIL?**

Title: Power Spectral Density Limit Fail

Description: This command queries the result of a limit check

on channel power spectral density

**:CALCulate:CHPower:LIMit:STATe <0 | 1 | ON | OFF>**
**:CALCulate:CHPower:LIMit:STATe?**

Title: Channel Power Limit State

Description: Sets and queries the state of limit on channel power. The set form of this command sets the limit state to be ON or OFF

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:CALCulate:CINTerference:CARrier:LIMit <numeric_value> {<amplitude_units>}**
**:CALCulate:CINTerference:CARrier:LIMit?**

Title: Carrier Limit

Description: Sets and queries the amplitude of the Carrier channel power limit for the Carrier-to-Interference Ratio measurement.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {<amplitude_units>}

Query Return: Numeric (<amplitude_units>)

Default Value: 10 dBm

Default Unit: dBm

Range: -150 dBm to 100 dBm

**:CALCulate:CINTerference:CARrier:LIMit:FAIL?**

Title: CI Carrier Limit Fail

Description: Queries the result of the Carrier channel power limit check for the Carrier-to-Interference Ratio measurement.

**:CALCulate:CINTerference:CARrier:LIMit:STATe <0 | 1 | ON | OFF>**
**:CALCulate:CINTerference:CARrier:LIMit:STATe?**

Title: Carrier Limit State

Description: Sets and queries the state of the Carrier channel power limit for the Carrier-to-Interference Ratio measurement.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:CALCulate:CINTerference:CIRatio:LIMit <numeric_value> {DB}**
**:CALCulate:CINTerference:CIRatio:LIMit?**

Title: CI Ratio Limit

Description: Sets and queries the amplitude of the Carrier-to-Interference Ratio measurement's relative limit.

Parameter(s): <numeric_value> {DB}

Query Return: Numeric (dB)

Default Value: 60 dB

Default Unit: dB

Range: 0 dB to 150 dB

**:CALCulate:CINTerference:CIRatio:LIMit:FAIL?**

Title: CI Ratio Limit Fail

Description: Queries the result of the Carrier-to-Interference ratio's relative limit check.

**:CALCulate:CINTerference:CIRatio:LIMit:STATe <0 | 1 | ON | OFF>**
**:CALCulate:CINTerference:CIRatio:LIMit:STATe?**

Title: CI Ratio Limit State

Description: Sets and queries the state of the Carrier-to-Interference Ratio measurement's relative limit. The set form of this command sets the limit state to be ON or OFF.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:CALCulate:CINTerference:INTerference:LIMit <numeric_value> {<amplitude_units>}**
**:CALCulate:CINTerference:INTerference:LIMit?**

Title: Interference Limit

Description: Sets and queries the amplitude of the Interference channel power limit for the Carrier-to-Interference Ratio measurement.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {<amplitude_units>}

Query Return: Numeric (<amplitude_units>)

Default Value: -50 dBm

Default Unit: dBm

Range: -150 dBm to 100 dBm

**:CALCulate:CINTerference:INTerference:LIMit:FAIL?**

Title: CI Interference Limit Fail

Description: Queries the result of the Interference channel power limit check for the Carrier-to-Interference Ratio measurement.

**:CALCulate:CINTerference:INTerference:LIMit:STATe <0 | 1 | ON | OFF>**
**:CALCulate:CINTerference:INTerference:LIMit:STATe?**

Title: Interference Limit State

Description: Sets and queries the state of the Interference channel power limit for the Carrier-to-Interference Ratio measurement. The set form of this command sets the limit state to be ON or OFF.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:CALCulate:EIRPower:HORizontal:LIMit:FAIL?**

Title: Horizontal EIRP Limit Pass/Fail

Description: This command queries the result of the Horizontal EIRP limit check. The query returns 1 for fail and 0 for pass.

**:CALCulate:EIRPower:HORizontal:SAVE**

Title: Save Horizontal EIRP

Description: Saves the active trace's EIRP value as the Horizontal EIRP. Recalculates Sum EIRP. Use :FETCh:EIRPower to query new Horizontal and SUM values.

**:CALCulate:EIRPower:LIMit:FAIL?**

Title: EIRP Limit Pass/Fail

Description: This command queries the result of a limit check on EIRP. The query returns 1 for fail and 0 for pass.

**:CALCulate:EIRPower:LIMit:LOWer <numeric_value> {<amplitude_units>}**
**:CALCulate:EIRPower:LIMit:LOWer?**

Title: EIRP Lower Limit

Description: This command sets or queries the EIRP Lower limit. The query returns the EIRP lower limit in the current amplitude unit.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {<amplitude_units>}

Query Return: Numeric (<amplitude_units>)

Default Value: 0 dBm

Default Unit: dBm

Range: -99 dBm to 100 dBm

**:CALCulate:EIRPower:LIMit:LOWer:STATe <0 | 1 | ON | OFF>**
**:CALCulate:EIRPower:LIMit:LOWer:STATe?**

Title: EIRP Lower Limit State

Description: This command sets or queries the EIRP lower limit state. The query returns 1 if the lower limit state is On, otherwise 0.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:CALCulate:EIRPower:LIMit:UPPer <numeric_value> {<amplitude_units>}**
**:CALCulate:EIRPower:LIMit:UPPer?**

Title: EIRP Upper Limit

Description: This command sets or queries the EIRP upper limit. The query returns the EIRP upper limit in the current amplitude unit.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {<amplitude_units>}

Query Return: Numeric (<amplitude_units>)

Default Value: 0 dBm

Default Unit: dBm

Range: -99 dBm to 100 dBm

**:CALCulate:EIRPower:LIMit:UPPer:STATe <0 | 1 | ON | OFF>**
**:CALCulate:EIRPower:LIMit:UPPer:STATe?**

Title: EIRP Upper Limit State

Description: This command sets or queries the EIRP upper limit state. The query returns 1 if the upper limit state is On, otherwise 0.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:CALCulate:EIRPower:MAXimum:LIMit:FAIL?**

Title: MAXimum EIRP Limit Pass/Fail

Description: This command queries the result of the maximum EIRP limit check. The query returns 1 for fail and 0 for pass.

**:CALCulate:EIRPower:SUM:LIMit:FAIL?**

Title: Sum EIRP Limit Pass/Fail

Description: This command queries the result of the SUM EIRP limit check. The query returns 1 for fail and 0 for pass.

**:CALCulate:EIRPower:VERTical:LIMit:FAIL?**

Title: Vertical EIRP Limit Pass/Fail

Description: This command queries the result of the Vertical EIRP limit check. The query returns 1 for fail and 0 for pass.

**:CALCulate:EIRPower:VERTical:SAVE**

Title: Save Vertical EIRP

Description: Saves the active trace's EIRP value as the Vertical EIRP. Recalculates Sum EIRP. Use :FETCh:EIRPower to query new Vertical and SUM values.

**:CALCulate:LIMit<n>:ENVelope:UPDate:Y**

Title: Update Limit Envelope Amplitude

Description: This command updates the amplitude of the upper and lower limits without changing the frequencies of the inflection points.

Note that this command is not available for Spectral Emission Mask measurement.

LIMit Suffix Range: 1-10, Default = 1

**:CALCulate:LIMit<n>:LOWer:ENVelope:CREate**

Title: Create Lower Limit Envelope

Description: This command is used to create an Lower limit envelope on the selected trace.

Note that this command is not available for Spectral Emission Mask measurement.

LIMit Suffix Range: 1-10, Default = 1

**:CALCulate:LIMit<n>:LOWer[:TRACe]:POINts?**

Title: Lower Limit Points

Description: This command queries the lower limit trace points (Y-axis values) for the specified limit. This differs from the limit data in that a value is returned for each point in the trace to indicate the values that are being used to evaluate limit pass/fail status.

The numeric suffix on LIMit specifies which limit number to query or set data. If the suffix is omitted, the command will refer to limit 1.

If a limit of the specified number does not exist, an empty list will be returned. Points which are not covered by the limit will be 'inf'.

Note that this command is not available for Spectral Emission Mask measurement.

LIMit Suffix Range: 1-10, Default = 1

**:CALCulate:LIMit<n>:UPPer:ENVelope:CREate**

Title: Create Upper Limit Envelope

Description: This command is used to create an Upper limit envelope on the selected trace.

Note that this command is not available for Spectral Emission Mask measurement.

LIMit Suffix Range: 1-10, Default = 1

**:CALCulate:LIMit<n>:UPPer[:TRACe]:POINts?**

Title: Upper Limit Points

Description: This command queries the upper limit trace points (Y-axis values) for the specified limit. This differs from the limit data in that a value is returned for each point in the trace to indicate the values that are being used to evaluate limit pass/fail status.

The numeric suffix on LIMit specifies which limit number to query or set data. If the suffix is omitted, the command will refer to limit 1.

If a limit of the specified number does not exist, an empty list will be returned. Points which are not covered by the limit will be 'inf'.

Note that this command is not available for Spectral Emission Mask measurement.

LIMit Suffix Range: 1-10, Default = 1

**:CALCulate:LIMit:ALARm <0 | 1 | ON | OFF>**
**:CALCulate:LIMit:ALARm?**

Title: Limit Alarm

Description: This command enables/disables the AAE notification for limit failures.

Note that this command is not available for Spectral Emission Mask measurement.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:CALCulate:LIMit:ENVelope:OFFSet <numeric_value> {<amplitude_units>}**
**:CALCulate:LIMit:ENVelope:OFFSet?**

Title: Limit Envelope Offset

Description: This command sets/gets the limit envelope offset. This defines how far away from the measured signal indicated the limit envelope is placed. Use :CALCulate:LIMit:TYPe to set the currently active limit line.

Note that this command is not available for Spectral Emission Mask measurement.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {<amplitude_units>}

Query Return: Numeric (<amplitude_units>)

Default Value: 3 dBm

Default Unit: dBm

Range: -100 dBm to 100 dBm

**:CALCulate:LIMit:ENVelope:POINt <numeric_value>**
**:CALCulate:LIMit:ENVelope:POINt?**

Title: Number of Limit Envelope Points

Description: This command sets the number of inflection point for the limit envelope.

Note that this command is not available for Spectral Emission Mask measurement.

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 20

Range: 2 to 41

**:CALCulate:LIMit:ENVelope:SHAPe <SQUare|SLOPe>**
**:CALCulate:LIMit:ENVelope:SHAPe?**

Title: Limit Envelope Shape

Description: This command sets/gets the currently active limit envelope shape.

Note that this command is not available for Spectral Emission Mask measurement.

Parameter(s): <SQUare|SLOPe>

Query Return: SQU|SLOP

Default Value: SQUare

**:CALCulate:MARKer<n>:FUNCtion <OFF|NOISe|FCOunter|QUASi>**
**:CALCulate:MARKer<n>:FUNCtion?**

Title: Marker Function

Description: Select the marker function to perform post-processing operation. The default/OFF function performs pass through operation.

Parameter(s): <OFF|NOISe|FCOunter|QUASi>

Query Return: OFF|NOIS|FCO|QUAS

MARKer Suffix Range: 1-12, Default = 1

Default Value: OFF

**:CALCulate:MARKer<n>:MAXimum**

Title: Maker Move To Highest Peak

Description: Moves the marker X value to the point in the marker's assigned trace that has the highest peak.

MARKer Suffix
Range: 1-12, Default = 1

**:CALCulate:MARKer<n>:MAXimum:LEFT**

Title: Marker Move To Left Peak

Description: Moves the marker X value to the point in the marker's assigned trace that is the highest peak to the LEFT of the current X position of the marker.

MARKer Suffix
Range: 1-12, Default = 1

**:CALCulate:MARKer<n>:MAXimum:NEXT**

Title: Marker Move To Next Peak

Description: Moves the marker X value to the point in the marker's assigned trace that is the next highest peak.

MARKer Suffix
Range: 1-12, Default = 1

**:CALCulate:MARKer<n>:MAXimum:RIGHt**

Title: Marker Move To Right Peak

Description: Moves the marker X value to the point in the marker's assigned trace that is the highest peak to the RIGHT of the current X position of the marker.

MARKer Suffix
Range: 1-12, Default = 1

**:CALCulate:MARKer<n>:MODE <POSition|DELTa|FIXed>**
**:CALCulate:MARKer<n>:MODE?**

Title: Marker Mode

Description: Select the marker mode. POSition mode is a normal marker mode. DELTa mode is marker report difference relative to another marker. FIXed mode is a marker who's Y value is fixed at the last measured value before it is set to FIXed.

Parameter(s): <POSition | DELTa | FIXed>

Query Return: POS | DELT | FIX

MARKer Suffix
Range: 1-12, Default = 1

Default Value: POSition

**:CALCulate:MARKer<n>:MOVE:LEFT**

Title: Marker Move To Left

Description: Moves the marker X value to the next display point to the LEFT.

MARKer Suffix
Range: 1-12, Default = 1

**:CALCulate:MARKer<n>:MOVE:RIGHt**

Title: Marker Move To Right

Description: Moves the marker X value to the next display point to the RIGHT.

MARKer Suffix Range: 1-12, Default = 1

**:CALCulate:MARKer<n>:REFerence <numeric_value>**
**:CALCulate:MARKer<n>:REFerence?**

Title: Marker Reference

Description: Sets the reference marker for the specified delta marker. This setting will be applied only if the marker mode set to DELTa.

Parameter(s): <numeric_value>

Query Return: Numeric

MARKer Suffix Range: 1-12, Default = 1

Default Value: 0

Range: 0 to 12

**:CALCulate:MARKer<n>[:SET]:CENTer**

Title: Marker Set Center

Description: Sets the center frequency of the instrument to the current X value of the marker.

MARKer Suffix Range: 1-12, Default = 1

**:CALCulate:MARKer<n>[:SET]:RLEVel**

Title: Marker Set Reference Level

Description: Sets the reference level of the instrument to the current Y value of the marker.

MARKer Suffix Range: 1-12, Default = 1

**:CALCulate:MARKer<n>:STATe <0 | 1 | ON | OFF>**
**:CALCulate:MARKer<n>:STATe?**

Title: Marker State

Description: Turns the selected marker on or off.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

MARKer Suffix Range: 1-12, Default = 1

Default Value: OFF

**:CALCulate:MARKer<n>:TRACe <numeric_value>**
**:CALCulate:MARKer<n>:TRACe?**

Title: Marker Trace
Description: Changes the trace to which the marker is currently attached.
Parameter(s): <numeric_value>
Query Return: Numeric
MARKer Suffix
Range: 1-12, Default = 1
Default Value: 1
Range: 1 to 6

**:CALCulate:MARKer<n>:X <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**:CALCulate:MARKer<n>:X?**

Title: Marker Position Zero Span
Description: Sets the marker time.
Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}
Query Return: Numeric (ms)
MARKer Suffix
Range: 1-12, Default = 1
Default Value: 29999.99 ms
Default Unit: ms
Range: -1000 ms to 61670 ms

**:CALCulate:MARKer<n>:X <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**:CALCulate:MARKer<n>:X?**

Title: Marker Position
Description: Sets the marker frequency.
Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}
Query Return: Numeric (Hz)
MARKer Suffix
Range: 1-12, Default = 1
Default Value: 200000000000 Hz
Default Unit: Hz
Range: -400000000000 Hz to 300000000000 Hz

**:CALCulate:MARKer<n>:Y <numeric_value> {<amplitude_units>}**
**:CALCulate:MARKer<n>:Y?**

Title: Marker Value

Description: Set/Get the marker Y value. The user defined Y maker value will be applied only if the marker is in Fixed mode.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {<amplitude_units>}

Query Return: Numeric (<amplitude_units>)

MARKer Suffix Range: 1-12, Default = 1

Default Value: 10 dBm

Default Unit: dBm

Range: -250 dBm to 130 dBm

**:CALCulate:MARKer:AOFF**

Title: Marker All Off

Description: Turns all markers off.

**:CALCulate:MARKer:APReset**

Title: Marker Preset

Description: Presets all markers to their preset values.

**:CALCulate:MARKer[:DATA]:ALL?**

Title: All Marker Data

Description: Returns the last stored X and Y values of all markers. The response format is (marker1.x, marker1.y), (marker2.x, marker2.y), ... (markern.x, markern.y). Returns (0,nan) if the marker has never been turned on or the marker is preset.

**:CALCulate:MARKer:FUNCtion:FCOunter:RESolution <R0|R1|R2|R3>**
**:CALCulate:MARKer:FUNCtion:FCOunter:RESolution?**

Title: Frequency Counter Marker Resolution

Description: This command specifies the desired counter marker frequency resolution.

Value Resolution

R0 1 Hz

R1 100 mHz

R2 10 mHz

R3 1 mHz

Parameter(s): <R0|R1|R2|R3>

Query Return: R0|R1|R2|R3

Default Value: R0

**:CALCulate:MARKer:PEAK:EXCursion <numeric_value> {DB}**
**:CALCulate:MARKer:PEAK:EXCursion?**

Title: Marker Excursion

Description: Sets the excursion for a marker. The excursion is the vertical distance from the peak to the next highest valley which must be exceeded for a peak to be considered a peak in marker max commands.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {DB}

Query Return: Numeric (dB)

Default Value: 0 dB

Default Unit: dB

Range: 0 dB to 200 dB

**:CALCulate:MARKer:PEAK:EXCursion:STATe <0 | 1 | ON | OFF>**
**:CALCulate:MARKer:PEAK:EXCursion:STATe?**

Title: Marker Excursion State

Description: Turn on/off excursion checking for marker max commands.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:CALCulate:MARKer:PEAK:THReshold <numeric_value> {<amplitude_units>}**
**:CALCulate:MARKer:PEAK:THReshold?**

Title: Marker Threshold

Description: Sets the threshold level for a marker. The threshold is the level that a peak must exceed to be considered a peak in marker max commands.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {<amplitude_units>}

Query Return: Numeric (<amplitude_units>)

Default Value: 0 dBm

Default Unit: dBm

Range: -200 dBm to 100 dBm

**:CALCulate:MARKer:PEAK:THReshold:STATe <0 | 1 | ON | OFF>**
**:CALCulate:MARKer:PEAK:THReshold:STATe?**

Title: Marker Threshold State

Description: Turn on/off threshold checking for marker max commands.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:CALCulate:OBW:LIMit <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**:CALCulate:OBW:LIMit?**

Title: OBW Limit

Description: Sets and queries the frequency limit on occupied bandwidth power.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

Query Return: Numeric (Hz)

Default Value: 10000 Hz

Default Unit: Hz

Range: 1000 Hz to 300000000000 Hz

**:CALCulate:OBW:LIMit:FAIL?**

Title: OBW Limit Fail

Description: This command queries the result of a limit check

on occupied bandwidth power

**:CALCulate:OBW:LIMit:STATe <0 | 1 | ON | OFF>**
**:CALCulate:OBW:LIMit:STATe?**

Title: OBW Power Limit State

Description: Sets and queries the state of limit on occupied bandwidth power. The set form of this command sets the limit state to be ON or OFF

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:CONFigure:ACPower**

Title: Configure Adjacent Channel Power Ratio

Description: Configures and enables the default adjacent channel power ratio measurement. Disables channel power and occupied bandwidth measurements. Sets the main channel bandwidth equal to the span. Sets the adjacent channel bandwidth and channel spacing equal to the main channel bandwidth. Sets the detection method to RMS. Sets the instrument to single sweep mode (:INITiate:CONTinuous OFF). Measurement settings can be modified by using the [:SENSe]:ACPower commands before initiating a sweep.

**:CONFigure:CHPower**

Title: Configure Channel Power

Description: Configures and enables the default channel power measurement. Disables ACPR and occupied bandwidth measurements. Sets the integration bandwidth equal to the span. Sets the detection method to RMS. Sets the instrument to single sweep mode (:INITiate:CONTinuous OFF). Measurement settings can be modified by using the [:SENSe]:CHPower commands before initiating a sweep. Note that this measurement is not valid in zero span.

**:CONFigure:CINTerference**

Title: Configure CI Measurement

Description: Configures and enables the default Carrier-to-Interference Ratio measurement. Sets the detection method to RMS. Sets the instrument to single sweep mode (:INITiate:CONTinuous OFF). Measurement settings can be modified by using the [:SENSe]:CINTerference commands before initiating a sweep. Note that this measurement is not valid in zero span.

**:CONFigure:EIRPower**

Title: Configure EIRP

Description: Configures the EIRP measurement. This will disable other Spectrum measurements, configure a normal peak detect trace, and set the instrument in a single sweep mode (:INITiate:CONTinuous: OFF). Measurement settings can be modified by using the [:SENSe]:EIRPower commands before initiating a sweep.

**:CONFigure:OBWidth**

Title: Configure Occupied Bandwidth

Description: Configures and enables the default occupied bandwidth measurement. Disables channel power and ACPR measurements. Sets the method to %. Sets the % of power to 99%. Sets the instrument to single sweep mode (:INITiate:CONTinuous: OFF). Measurement settings can be modified by using the [:SENSe]:OBWidth commands before initiating a sweep. Note that this measurement is not valid in zero span.

**:CONFigure:SEMask**

Title: Configure Spectral Emission Mask Measurement

Description: Configures and enables the Spectral Emission Mask measurement. This will disable all other measurements, set the span to mask width, set the detection method to RMS, and set the instrument in a single sweep mode (:INITiate:CONTinuous OFF). Measurement settings can be modified by using the [:SENSe]:SEMask commands before initiating a sweep.

**:CONFigure:USER:CARRiers[:STATe] <0 | 1 | ON | OFF>**
**:CONFigure:USER:CARRiers[:STATe]?**

Title: SIB1 Carrier User Configuration State

Description: Enable or disable custom user configuration of mcc, mnc pairs to mobile network names mapping.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:CONFigure:USER:COUNtries[:STATe] <0 | 1 | ON | OFF>**
**:CONFigure:USER:COUNtries[:STATe]?**

Title: SIB1 Country User Configuration State

Description: Enable or disable custom user configuration of mcc codes to country names mapping.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:CSCanner:ADD:CUSTom <numeric_value> {HZ | KHZ | MHZ | GHZ},<numeric_value> {HZ | KHZ | MHZ | GHZ},[<numeric_value>],[<string>],[<numeric_value> {PS | NS | US | MS |**

**S | MIN | HR}],[<numeric_value> {<amplitude_units>}],[<numeric_value> {<amplitude_units>}]**

Title: Add a Custom Channel

Description: This command adds a new channel to the channel scanner mode list, created from a user specified Center Frequency and Channel Span.

Parameters:

1. Center Freqency: Center fequency of the new channel.

2. Channel Span: The delta between the start frequency and stop frequency of the new channel.

3. Index: Channel scanner list index of the new channel.

(optional)

4. Channel Name: User specified channel name. (optional)

5. Dwell Time: The time duration spent scanning a given channel, specified for all channels copied. A 0 or no value indicates measurement complete fast as possible. (optional)

6. Upper Limit: Channel power upper limit (in dBm), specified for all channels copied. In absence of a user specified limit, infinite power will be used, and the limit will always pass. (optional)

7. Lower Limit: Channel power lower limit (in dBm), specified for all channels copied. In absence of a user specified limit, zero power will be used, and the limit will always pass. (optional)

For example:

CSC:ADD:CUST 890.3 MHz, 200 KHz, 0, "Test", 500 ms, -20dbm, -70dbm

The channel scanner list will be populated with a single new channel, with the user supplied Center Frequency and Channel Span. The new channels is inserted at the user specified Index, 0. The channel will be labled with a name, in this case TEST. Time spent on the new channel will be no less than, and as close as possible to, the user supplied Dwell Time, in this case 500 ms. The channel features an upper limit on its measured channel power of -20 dBm, and a lower limit on its measured channel power of -70 dBm. NOTE:

Channel Creation Limits:

Channels may only be generated within the frequency span of the analyzer. For a given command issuance, if the channel span generated by user supplied Center Freqency and Channel Span results in either: Center Frequency - (1/2 * Channel Span) < Analyzer Min Frequency

Center Frequency + (1/2 * Channel Span) > Analyzer Max Frequency

such a channel will not be generated.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ},<numeric_value> {HZ | KHZ | MHZ | GHZ},[<numeric_value>],[<string>],[<numeric_value> {PS | NS | US | MS | S | MIN | HR}],[<numeric_value> {<amplitude_units>}],[<numeric_value> {<amplitude_units>}]

**:CSCanner:ADD:FREQuency <numeric_value> {HZ | KHZ | MHZ | GHZ},<numeric_value> {HZ | KHZ | MHZ | GHZ},<numeric_value> {HZ | KHZ | MHZ | GHZ},<numeric_value>,[<numeric_value>],[<string>],[<numeric_value> {PS**

**| NS | US | MS | S | MIN | HR}],[<numeric_value> {<amplitude_units>}],[<numeric_value> {<amplitude_units>}]**

Title: Add Channels from Frequency Range

Description: This command adds new channels to the channel scanner mode's list, created from a user specified frequency range.

Parameters:

1. Start Freqency: Start frequency of the first channel's span.

2. Channel Spacing: The delta between adjacent channel start frequencies.

3. Channel Span: The delta between the start frequency and stop frequency of channels created.

4. Channel Count: Number of channels to be created.

5. Index: Channel scanner list index to be used for the created channel of lowest start frequency, with subsequent (Channel Count - 1) channels enumerated sequentially. (optional)

6. Channel Name: User specified channel name. (optional)

7. Dwell Time: The time duration spent scanning a given channel, specified for all channels copied. A 0 or no value indicates measurement complete fast as possible. (optional)

8. Upper Limit: Channel power upper limit (in dBm), specified for all channels copied. In absence of a user specified limit, infinite power will be used, and the limit will always pass. (optional)

9. Lower Limit: Channel power lower limit (in dBm), specified for all channels copied. In absence of a user specified limit, zero power will be used, and the limit will always pass. (optional)

For example:

CSC:ADD:FREQ 890.2 MHz, 200 KHz, 200 KHz, 5, 0, "Test", 500 ms, -20dBm, -70dBm

The channel scanner list will be populated with channels, starting at the user specified Start Frequency, in this case 890.2 MHz. Channels created will span frequencies equal to the user specified Channel Span, in this case 200 KHz. Subsequent channels generated will have adjacent spans, with the delta between adjacent channel's Start Frequency equal to the specified Channel Spacing, in this case 200 KHz. As Channel Spacing is equal to Channel Span, the channels created by this command invocation will be contiguous. However, users may generate channels with overlapping spans or channels with gaps or guard bands between channels. The number of channels generated will be equal to the user specified Channel Count, in this case 5. Thus channel's will be created with center frequencies: 890.3, 890.5, 890.7, 890.9, 891.1 (MHz) The 5 new channels are inserted at the user specified Index, 0, and are enumerated sequentially through index 4. The channel will be labled with a name, in this case TEST. Time spent on each created channel will be no less than, and as close as possible to, the user supplied Dwell Time, in this case 500 ms. Each new channel features an upper limit on their measured channel power of -20 dBm, and a lower limit on their measured channel power of -70 dBm.

NOTE:

Channel Creation Limits:

Channels may only be generated within the frequency span of the analyzer. For a given command issuance, if the Nth channel to be generated by user supplied Start Frequency, Channel Spacing, and Channel Span results in a channel for which: Start Frequency < Analyzer Min Frequency

Start Frequency + (N * Channel Spacing) > Analyzer Max Frequency

Start Frequency + Channel Span + (N * Channel Spacing) > Analyzer Max Frequency

such channels will not be generated regardless of the user supplied Channel Count. However all created channels with a valid span will be added to the channel scanner list.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ},<numeric_value> {HZ | KHZ | MHZ | GHZ},<numeric_value> {HZ | KHZ | MHZ | GHZ},<numeric_value>,[<numeric_value>],[<string>],[<numeric_value> {PS | NS | US | MS | S | MIN | HR}],[<numeric_value> {<amplitude_units>}],[<numeric_value> {<amplitude_units>}]

**:CSCanner:ADD:SIGnalstandard <string>,<numeric_value>,<numeric_value>,<numeric_value>,[<numeric_value>],[<numeric_value> {HZ | KHZ | MHZ | GHZ}],[<numeric_value> {PS | NS**

**| US | MS | S | MIN | HR}],[<numeric_value> {<amplitude_units>}],[<numeric_value> {<amplitude_units>}]**

Title: Add Channels from a Signal Standard

Description: This command adds new channels to the channel scanner mode's list, copied from a user selected signal standard band.

Parameters:

1. Signal Standard: Name of the signal standard band to copy channel information from.

2. Start Channel: The first channel index to be copied from the named Signal Standard.

3. Channel Step Size: The delta between two channels copied from the named Signal Standard.

4. Channel Count: Number of channels intended to be copied.

5. Index: Channel scanner list insertion index to be used for the copied channel of lowest start frequency, with subsequent (Channel Count - 1) channels enumerated sequentially. (optional)

6. Channel Span: A value to override the signal standard ChannelSpan attribute. No value will use the signal standard's default value. (optional)

7. Dwell Time: The time duration spent scanning a given channel, specified for all channels copied. A 0 or no value indicates measurement complete fast as possible. (optional)

8. Upper Limit: Channel power upper limit (in dBm), specified for all channels copied. In absence of a user specified limit, infinite power will be used, and the limit will always pass. (optional)

9. Lower Limit: Channel power lower limit (in dBm), specified for all channels copied. In absence of a user specified limit, zero power will be used, and the limit will always pass. (optional)

For example:

CSC:ADD:SIG "P-GSM 900 - Uplink", 1, 3, 5, 0, 20 MHz, 500 ms, -20dBm, -70dBm

The channel scanner list will be populated with channels from the named Signal Standard band, in this case P-GSM 900 - Uplink. Channels created will copy selected channel's channel span and center frequency, beginning with the user specified Start Channel, in this case Signal Standard channel 1. As this band's channel step is 1, with the user supplied Channel Step Size, 3, the remaining Signal Standard channels to be copied are 4, 7, 10, and 13. As this band's channels begin with start frequency 890.2 MHz and this band's channel spacing is 200 KHz, channel's will be created with center frequencies: 890.3, 890.9, 891.5, 892.1, 892.7 (MHz) The 5 new channels are inserted at the user specified Index, 0, and are enumerated sequentially through index 4.

The channel's span will be set to the user provided value of 20 MHz. Time spent on each copied channel will be no less than, and as close as possible to, the user supplied Dwell Time, in this case 500 ms. Each new channel features an upper limit on their measured channel power of -20 dBm, and a lower limit on their measured channel power of -70 dBm. NOTE:

Channel Creation Limits:

Channels may only be generated from a Start Channel valid for the named Signal Standard band. For a given command issuance, if the user supplied Start Channel, Channel Step Size, and Channel Count result in channel numbers that either preceed or exceed the named Signal Standard band's channel numbers, then such channels are not created. However all channels valid for the named Signal Standard will be added to the channel scanner list.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s):

<string>,<numeric_value>,<numeric_value>,<numeric_value>,[<numeric_value>],[<numeric_value> {HZ | KHZ | MHZ | GHZ}],[<numeric_value> {PS | NS | US | MS | S | MIN | HR}],[<numeric_value> {<amplitude_units>}],[<numeric_value> {<amplitude_units>}]

**:CSCanner:AVERage:TYPE <LAST|MAX|MIN|AVER|RMAX|RMIN|RAV>**
**:CSCanner:AVERage:TYPE?**

Title: Channel Scanner Averaging Type

Description: Set the channel averaging type to indicate the field for determining the average against.

Channel Parameter:

LAST: Current

MAX: Channel Power Maximum

MIN: Channel Power Minimum

AVER: Channel Power Average

RMAX: Channel Power Rolling Maximum

RMIM: Channel Power Rolling Minimum

RAV: Channel Power Rolling Average

Parameter(s): <LAST | MAX | MIN | AVER | RMAX | RMIN | RAV>

Query Return: LAST | MAX | MIN | AVER | RMAX | RMIN | RAV

Default Value: LAST

**:CSCanner:CHANnel?**

Title: Channel List

Description: Query the parameters of all channels in Channel Scanner list.

**:CSCanner:CHANnel:LIMit?**

Title: Channel Member Limit

Description: Query the maximum Channel Scanner list size.

Query Return: Numeric

Default Value: 60

Range: 60 to 60

**:CSCanner:CLEar <numeric_value>**

Title: Clear Channel

Description: Clear all measurement results of a channel at a given Channel Scanner list index.

Parameter(s): <numeric_value>

**:CSCanner:CLEar:ALL**

Title: Clear All Channels

Description: Clear all measurement results of all channels in Channel Scanner list.

**:CSCanner:DELete <numeric_value>**

Title: Delete Channel

Description: Delete a channel at a given Channel Scanner list index

Parameter(s): <numeric_value>

**:CSCanner:DELete:ALL**

Title: Delete All Channels

Description: Delete all channels in Channel Scanner list.

**:CSCanner:EDIT:CUSTom <numeric_value> {HZ | KHZ | MHZ | GHZ},<numeric_value> {HZ | KHZ | MHZ | GHZ},<numeric_value>,[<string>],[<numeric_value> {PS | NS | US | MS | S | MIN | HR}],[<numeric_value> {<amplitude_units>}],[<numeric_value> {<amplitude_units>}]**

Title: Edit Channel with custom data

Description: This command replaces a member of the channel scanner mode's list, created from user specified Center Frequency and Channel Span.

Parameters:

1. Center Freqency: Center fequency of the new channel.

2. Channel Span: The delta between the start frequency and stop frequency of the new channel.

3. Index: Channel scanner list index of the channel to be replaced.

4. Channel Name: User specified channel name. (optional)

5. Dwell Time: The time duration spent scanning a given channel, specified for all channels copied. A 0 or no value indicates measurement complete fast as possible. (optional)

6. Upper Limit: Channel power upper limit (in dBm), specified for all channels copied. In absence of a user specified limit, infinite power will be used, and the limit will always pass. (optional)

7. Lower Limit: Channel power lower limit (in dBm), specified for all channels copied. In absence of a user specified limit, zero power will be used, and the limit will always pass. (optional)

For an example of this channel creation method, see CSCanner:ADD:CUSTom.

The channel created by this command is done in the same way and with the same restrictions.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ},<numeric_value> {HZ | KHZ | MHZ | GHZ},<numeric_value>,[<string>],[<numeric_value> {PS | NS | US | MS | S | MIN | HR}],[<numeric_value> {<amplitude_units>}],[<numeric_value> {<amplitude_units>}]

**:CSCanner:EDIT:SIGnalstandard <string>,<numeric_value>,<numeric_value>,[<numeric_value> {HZ | KHZ | MHZ | GHZ}],[<numeric_value> {PS | NS | US | MS | S | MIN |**

**HR}],[<numeric_value> {<amplitude_units>}],[<numeric_value> {<amplitude_units>}]**

Title: Edit Channel with Signal Standard data

Description: This command replaces a member of the channel scanner mode's list, using data copied from a user selected signal standard band.

Parameters:

1. Signal Standard: Name of the signal standard band to copy channel information from.

2. Channel Index: The index of the channel to be copied from the named Signal Standard.

3. Index: Channel scanner list index of the channel to be replaced.

4. Channel Span: A value to override the signal standard ChannelSpan attribute. No value will use the signal standard's default value. (optional)

5. Dwell Time: The time duration spent scanning a given channel, specified for all channels copied. A 0 or no value indicates measurement complete fast as possible. (optional)

6. Upper Limit: Channel power upper limit (in dBm), specified for all channels copied. In absence of a user specified limit, infinite power will be used, and the limit will always pass. (optional)

7. Lower Limit: Channel power lower limit (in dBm), specified for all channels copied. In absence of a user specified limit, zero power will be used, and the limit will always pass. (optional)

For an example of this channel creation method, see CSCanner:ADD:SIGnalstandard.

The channel copied by this command is done in the same way and with the same restrictions.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <string>,<numeric_value>,<numeric_value>,[<numeric_value> {HZ | KHZ | MHZ | GHZ}],[<numeric_value> {PS | NS | US | MS | S | MIN | HR}],[<numeric_value> {<amplitude_units>}],[<numeric_value> {<amplitude_units>}]

**:CSCanner:LOG <0 | 1 | ON | OFF>**
**:CSCanner:LOG?**

Title: Channel Scanner Logging Enable

Description: Set or query whether or not Channel Scanner data logging is enabled.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:CSCanner:LOG:INTerval <numeric_value> {PS | NS | US | MS | S | MIN | HR}**

**:CSCanner:LOG:INTerval?**

Title: Channel Scanner Logging Interval Time

Description: Set or query the interval time between logging data point.

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 1000 ms

Default Unit: ms

Range: 100 ms to 60000 ms

**:CSCanner:RESults? [<ALL|STRip|BAR>],[<numeric_value>]**

Title: Channel Results

Description: Query channel results with options to filter on result type and channel index.

Parameters:

1. Result Type: Enumerable value specifying the result type to query. ALL or the lack of a result type returns all channel data, which includes: channel creation parameters channel history

channel measurement result data index of the most recently scanned channel current channel list size. STRip returns channel history, channel creation parameters, last scanned channel index, and list size. BAR returns channel measurement results, creation parameters, last scanned channel index, and list size. (optional)

2. Index: Channel scanner list index to query measurement results from. Leaving this field empty returns results from every channel. (optional)

Parameter(s): [<ALL|STRip|BAR>],[<numeric_value>]

**:CSCanner:SORT:ASC <CFR|BWID|DTIM|LIMI|LAST|MAX|MIN|AVER|RMAX|RMIN|RAV>**

Title: Sort Channels in Ascending Order

Description: Sort the channels in ascending order based on the selected channel parameter or result data.

Channel Parameter/Result:

CFR: Center Frequency

BWID: Integration Bandwidth

DTIM: Dwell Time

LIMI: Limit Pass/Fail

LAST: Channel Power Last Value

MAX: Channel Power Maximum

MIN: Channel Power Minimum

AVER: Channel Power Average

RMAX: Channel Power Rolling Maximum

RMIM: Channel Power Rolling Minimum

RAV: Channel Power Rolling Average

Parameter(s): <CFR|BWID|DTIM|LIMI|LAST|MAX|MIN|AVER|RMAX|RMIN|RAV>

**:CSCanner:SORT:DESC**
**<CFR|BWID|DTIM|ULIM|LLIM|LAST|MAX|MIN|AVER|RMAX|RMIN|RAV>**

Title: Sort Channels in Descending Order

Description: Sort the channels in descending order based on the selected channel parameter or result data.

Channel Parameter/Result:

CFR: Center Frequency

BWID: Integration Bandwidth

DTIM: Dwell Time

ULIM: Upper Limit Pass/Fail

LLIM: Lower Limit Pass/Fail

LAST: Channel Power Last Value

MAX: Channel Power Maximum

MIN: Channel Power Minimum

AVER: Channel Power Average

RMAX: Channel Power Rolling Maximum

RMIM: Channel Power Rolling Minimum

RAV: Channel Power Rolling Average

Parameter(s): <CFR | BWID | DTIM | ULIM | LLIM | LAST | MAX | MIN | AVER | RMAX | RMIN | RAV>

**:CSCanner:SSTandard:ATTRibutes? <string>**

Title: Signal Standard Attributes Query

Description: This command returns the attributes associated with a named signal standard. Such attributes are useful for reference when parameterizing CSCanner:ADD:SIGnalstandard command invocations. Attributes:

ChannelSpan: The span of each channel in the band

Increment: The delta between the bands channel numbers

Spacing: The delta beteween start frequencies

StartChannel: The first channel number in the band

StartFrequency: The start frequency of the first channel in the band

StopChannel: The last channel number in the band

StopFrequency: The stop frequency of the last channel in the band

See CSCanner:SSTandard:CATalog for a list of supported signal standard names.

Parameter(s): <string>

**:CSCanner:SSTandard:CATalog?**

Title: Signal Standard Catalog Query

Description: This command returns a list of available signal standards as a json array Example return value:

{ "Signal Standards": [ "P-GSM 900 - Uplink", "P-GSM 900 - Downlink", "DCS 1800 - Uplink", "DCS 1800 - Downlink", ... (cont.)

**:CSCanner:UPDate:STATe <0 | 1 | ON | OFF>**
**:CSCanner:UPDate:STATe?**

Title: Channel Scanner Update

Description: Channel update state determines whether the channel scanner is update in every sweep.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**:DIAGnostic:SWEep:TIME?**

Title: Measured Sweep Time

Description: This command queries the measured sweep time, in number of milliseconds. This command will return "nan" if no measured sweep time is available, which happens if the sweep was reset and the instrument has not yet swept enough to measure a full sweep.

**:DISPlay:EXTernal:DEVices?**

Title: External devices

Description: Gets a json object with the external devices configuration.

**:DISPlay:IMPedance:MATCh:DEVice <NONe|M75|OTHer>**
**:DISPlay:IMPedance:MATCh:DEVice?**

Title: Impedance matching device

Description: Sets the impedance matching device.

This can be used to compensate for the insertion loss

of an external impedance matching device such as the

Anritsu 12N50-75B adapter.

NONe: Default 50 Ohm match (no adapter)

M75: Scalar correction for Anritsu 12N50-75B (75 Ohm to 50 Ohm adapter)

OTHer: Alows the user to enter a custom gain value

Parameter(s): <NONe|M75|OTHer>

Query Return: NON|M75|OTH

Default Value: NONe

**:DISPlay:IMPedance:MATCh:DEVice:CUSTom:LOSS <numeric_value> {DB}**
**:DISPlay:IMPedance:MATCh:DEVice:CUSTom:LOSS?**

Title:

Description: Sets custom loss value for when the impedance matching device is "OTHer"

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {DB}

Query Return: Numeric (dB)

Default Value: 0 dB

Default Unit: dB

Range: -99.9 dB to 99.9 dB

**:DISPlay:POINtcount <numeric_value>**
**:DISPlay:POINtcount?**

Title: Display Point Count

Description: Changes the number of display points the instrument currently measures. Increasing the number of display points can improve the resolution of measurements but will also increase sweep time.

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 501

Range: 10 to 4001

**:DISPlay:VIEW:SPECtrogram:AADJust**

Title: Spectrogram Scale Auto Adjust

Description: Automatically adjusts the spectrogram scale reference and bottom amplitude to the max and min amplitudes, respectively, of the most recent trace data collected.

**:DISPlay:VIEW:SPECtrogram:BOTTom <numeric_value> {<amplitude_units>}**
**:DISPlay:VIEW:SPECtrogram:BOTTom?**

Title: Spectrogram Scale Bottom

Description: The spectrogram scale bottom indicates the amplitude that corresponds to the least intense color in the spectrogram waterfall display. This is used in conjunction with the spectrogram scale reference command to establish the color gradient scale.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {<amplitude_units>}

Query Return: Numeric (<amplitude_units>)

Default Value: -90 dBm

Default Unit: dBm

Range: -150 dBm to 30 dBm

**:DISPlay:VIEW:SPECtrogram:HISTory[:DATA]? <numeric_value>**

Title: Spectrogram History Data

Description: This command transfers the spectrogram history from the instrument to the client.

This command takes an integer parameter specifying the last trace ID received by client and returns all trace data after the received trace ID in a binary block response. The binary block response is formatted as below

History Header: (20 bytes)

Number Of Traces: 4 bytes, unsigned integer

Number Of Trace Points Per Trace : 4 bytes, unsigned integer

Last trace ID in the block: 4 bytes, unsigned integer

Reserved: 8 bytes

History Body:

Block of trace points formatted as a 4 byte, single-precision floating point value for each point. The history body size (in bytes) can be calculated as: <Number Of Traces> * (<Number Of Trace Points Per Trace> * 4).

If no new trace data was found in history, this command will return header only binary block with all zeros except <last trace ID in the block> is set to the last trace ID received by client.

Parameter(s): <numeric_value>

**:DISPlay:VIEW:SPECtrogram:HISTory:ELAPsed? <numeric_value>**

Title: Spectrogram History Elapsed

Description: Provides the time values for the specified index in spectrogram history. Returns a negative value if the specified index is invalid.

Parameter(s): <numeric_value>

**:DISPlay:VIEW:SPECtrogram:HUE <numeric_value>**
**:DISPlay:VIEW:SPECtrogram:HUE?**

Title: Spectrogram Hue

Description: The spectrogram hue command indicates the color that represents the highest amplitude in the spectrogram waterfall display. All of the other amplitudes are represented by colors starting at this hue setting and following a color wheel.

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 0

Range: 0 to 359

**:DISPlay:VIEW:SPECtrogram:REFerence <numeric_value> {<amplitude_units>}**
**:DISPlay:VIEW:SPECtrogram:REFerence?**

Title: Spectrogram Scale Reference

Description: The spectrogram scale reference indicates the amplitude that corresponds to the most intense color in the spectrogram waterfall display. This is used in conjunction with the spectrogram scale bottom command to establish the color gradient scale.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {<amplitude_units>}

Query Return: Numeric (<amplitude_units>)

Default Value: 10 dBm

Default Unit: dBm

Range: -150 dBm to 30 dBm

**:DISPlay:VIEW:SPECtrogram:TRACe:SELection <POSition|TIME>**
**:DISPlay:VIEW:SPECtrogram:TRACe:SELection?**

Title: Cursor Selection

Description: Selects which time cursor setting to use, position or time. For position, the DISP:VIEW:SPEC:POS setting will be used to determine which trace from the spectrogram history to show. For time, the DISP:VIEW:SPEC:TIME setting will be used instead.

Parameter(s): <POSition|TIME>

Query Return: POS|TIME

Default Value: POSition

**:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision <numeric_value>**
**:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision?**

Title: Scale Per Division

Description: Set or query the scale per division setting of trace graph. This command doesn't change any behavior in the backend, but will be included in save/recall operations.

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 10

Range: 1 to 15

**:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision <numeric_value>**
**:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision?**

Title: Scale Per Division

Description: Set or query the scale per division setting of trace graph. This command doesn't change any behavior in the backend, but will be included in save/recall operations.

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 10

Range: 1 to 15

**:DISPlay[:WINDow]:TRACe:Y:SCALe:RLEVel <numeric_value> {<amplitude_units>}**
**:DISPlay[:WINDow]:TRACe:Y:SCALe:RLEVel?**

Title: Reference Level

Description: Sets the reference level amplitude value for the y-axis. This value is the display reference level, which means it has the reference level offset applied. It also means that a change to the reference level offset will change this setting (though the actual, unadjusted reference level will stay the same).

Note that this may cause a change in attenuation if the automatic input attenuation coupling is enabled. For the purpose of coupling, the actual reference level (without the offset applied) is used, though this command will always reflect the display value.

For example, suppose a starting reference level offset of 0 dB and a reference level of 10 dBm. If the offset is set to 20 dB, the display reference level will be set to 10 dBm - 20 dB = -10 dBm; no attenuation change will occur. Likewise, if the starting reference level offset is 20 dB and the (display) reference level is set to 0 dBm, the display reference level will remain at 0 dBm, though the (actual) reference level value used in auto attenuation coupling is 0 dBm + 20 dB = 20 dBm.

To obtain the actual reference level, either add in the reference level offset, or temporarily set the offset to 0 (in which case the display reference level will reflect the actual one).

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {<amplitude_units>}

Query Return: Numeric (<amplitude_units>)

Default Value: 10 dBm

Default Unit: dBm

Range: -150 dBm to 30 dBm

**:DISPlay[:WINDow]:TRACe:Y[:SCALe]:RLEVel:AUTO[:IMMediate] <numeric_value> {DB}**

Title: Automatic Reference Level

Description: This command causes the reference level to immediately be set to an automatically computed value that best displays the particular data. The reference level is set to a specified dB above the selected trace max value. This command is semantically equivalent to :DISPlay[:WINDow]:TRACe[:SCALe]:RLEVel <amplitude> with the computed value for amplitude.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {DB}

**:DISPlay[:WINDow]:TRACe:Y[:SCALe]:RLEVel:OFFSet <numeric_value> {DB}**
**:DISPlay[:WINDow]:TRACe:Y[:SCALe]:RLEVel:OFFSet?**

Title: Reference Level Offset

Description: Sets the reference level offset value for the y-axis.

This offset is used for display purposes only, and does not affect the actual reference level used for auto attenuation coupling, or any other settings that depend on reference level.

See DISP:WIND:TRAC:Y:SCAL:RLEV for details on the interaction between reference level offset and display reference level.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {DB}

Query Return: Numeric (dB)

Default Value: 0 dB

Default Unit: dB

Range: -99.9 dB to 99.9 dB

**:DISPlay[:WINDow]:TRACe:Y[:SCALe]:RLEVel:OFFSet <numeric_value> {DB}**
**:DISPlay[:WINDow]:TRACe:Y[:SCALe]:RLEVel:OFFSet?**

Title: Reference Level Offset

Description: Sets the reference level offset value for the y-axis.

This offset is used for display purposes only, and does not affect the actual reference level used for auto attenuation coupling, or any other settings that depend on reference level.

See DISP:WIND:TRAC:Y:SCAL:RLEV for details on the interaction between reference level offset and display reference level.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {DB}

Query Return: Numeric (dB)

Default Value: 0 dB

Default Unit: dB

Range: -99.9 dB to 99.9 dB

**:DISPlay[:WINDow]:TRACe:Y:SSCale:PDIVision <numeric_value> {DB}**
**:DISPlay[:WINDow]:TRACe:Y:SSCale:PDIVision?**

Title: Secondary Scale Per Division

Description: Set or query the scale per division setting the secondary axis scale (E.G. For trace math trace).

Parameter(s): <numeric_value> {DB}

Query Return: Numeric (dB)

Default Value: 10 dB

Default Unit: dB

Range: 1 dB to 15 dB

**:DISPlay[:WINDow]:TRACe:Y:SSCale:PDIVision <numeric_value> {DB}**
**:DISPlay[:WINDow]:TRACe:Y:SSCale:PDIVision?**

Title: Secondary Scale Per Division

Description: Set or query the scale per division setting the secondary axis scale (E.G. For trace math trace).

Parameter(s): <numeric_value> {DB}

Query Return: Numeric (dB)

Default Value: 10 dB

Default Unit: dB

Range: 1 dB to 15 dB

**:DISPlay[:WINDow]:TRACe:Y:SSCale:RLEVel <numeric_value> {DB}**
**:DISPlay[:WINDow]:TRACe:Y:SSCale:RLEVel?**

Title: Secondary Reference Level

Description: Sets the reference level amplitude value for the secondary y-axis. (E.G. For trace math trace)

Parameter(s): <numeric_value> {DB}

Query Return: Numeric (dB)

Default Value: 10 dB

Default Unit: dB

Range: -100 dB to 100 dB

**:DISPlay[:WINDow]:TRACe:Y:SSCale:RLEVel <numeric_value> {DB}**
**:DISPlay[:WINDow]:TRACe:Y:SSCale:RLEVel?**

Title: Secondary Reference Level

Description: Sets the reference level amplitude value for the secondary y-axis. (E.G. For trace math trace)

Parameter(s): <numeric_value> {DB}

Query Return: Numeric (dB)

Default Value: 10 dB

Default Unit: dB

Range: -100 dB to 100 dB

**:FETCh:ACPower?**

Title: Fetch Adjacent Channel Power Ratio

Description: Returns the most recent adjacent channel power ratio measurement results. If the instrument is sweeping, it will not return until the sweep is complete. If the instrument is not sweeping and the current data is not valid it will return error -230. This could occur if there was a *RST immediately before the :FETCh? or if a measurement parameter was changed without an :INITiate

Data is returned as 9 comma-separated values: main channel power, absolute lower adjacent channel power, absolute upper adjacent channel power, absolute lower alternate channel power, absolute upper alternate channel power, relative lower adjacent channel power, relative upper adjacent channel power, relative lower alternate channel power, relative upper alternate channel power.

If the measurement is not enabled (see [:SENSe]:CHPower:STATe) then, the instrument will indicate error -400 and return the string "nan,nan,nan,nan,nan,nan,nan,nan,nan"

**:FETCh:AMPLitude? <numeric_value> {HZ | KHZ | MHZ | GHZ}**

Title: Fetch Amplitude

Description: Returns the amplitude at the given frequency. The command does not wait for the sweep to complete.

If the trace data at the requested frequency is invalid (or out of span) then NAN is returned and error code -230 is indicated.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

**:FETCh:BPOWer?**

Title: Fetch Burst Average Power

Description: Returns the most recent burst average power measurement result. If the measurement is not enabled (see [:SENSe]:BPOWer:STATe) then, the instrument will indicate error -400. If a measurement is in progress, it blocks until complete.

**:FETCh:CAGGregation?**

Title: Fetch Carrier Aggregation Measurements

Description: Returns the Carrier Aggregation measurement results for each carrier in a JSON format. Each carrier measurement result will include the Carrier Id, measurement name, and measurement results. If a measurement is in progress, it blocks until the measurement is completed. Data is available only up to the available number of carriers, the rest will be an empty JSON. See command [:SENSe]:CAGGregation:CARRier:COUNt to set or query the number of carriers. An example of Carrier Aggregation measurement in 5GNR is as follows (Please refer to the individual measurement for further example):

{

"CarrierAggregationResults":

[{"CarrierId":1,

"PciMeasurement":[{...}],

"SsbMeasurement":[{...]},

]},

{},{},{},{},{},{},{}]}

**:FETCh:CHPower?**

Title: Fetch Channel Power and Density

Description: This command returns the most recent channel power measurement results: channel power and channel power density.

If the instrument is sweeping, it will not return until the sweep is complete. If the instrument is not sweeping and the current data is not valid it will return error -230. This could occur if there was a *RST immediately before the :FETCh? or if a measurement parameter was changed without an :INITiate. Data is returned as 2 comma-separated values: channel power, channel power density.

If the measurement is not enabled (see [:SENSe]:CHPower:STATe) then, the instrument will indicate error -400 and return the string "nan,nan"

**:FETCh:CHPower:CHPower?**

Title: Fetch Channel Power

Description: Returns the most recent channel power measurement result. It returns only the channel power, not the channel power density. Use :FETCh:CHPower? to get both channel power and channel power density.

If the instrument is sweeping, it will not return until the sweep is complete. If the instrument is not sweeping and the current data is not valid it will return error -230. This could occur if there was a *RST immediately before the :FETCh? or if a measurement parameter was changed without an :INITiate.

If the measurement is not enabled (see [:SENSe]:CHPower:STATe) then, the instrument will indicate error -400 and return the string "nan,nan"

**:FETCh:CHPower:DENSity?**

Title: Fetch Channel Power Density

Description: Returns the most recent channel power density measurement result. It returns only the channel power density, not the channel power.

Use :FETCh:CHPower? to get both channel power and channel power density.

If the instrument is sweeping, it will not return until the sweep is complete. If the instrument is not sweeping and the current data is not valid it will return error -230. This could occur if there was a *RST immediately before the :FETCh? or if a measurement parameter was changed without an :INITiate.

If the measurement is not enabled (see [:SENSe]:CHPower:STATe) then, the instrument will indicate error -400 and return the string "nan,nan"

**:FETCh:CINTerference?**

Title: Fetch CI Ratio

Description: Returns the most recent Carrier-to-Interference measurement result. If the instrument is sweeping, it will not return until the sweep is complete. If the instrument is not sweeping and the current data is not valid it will return error -230. This could occur if there was a *RST immediately before the :FETCh?

If the measurement is not enabled (see [:SENSe]:CINTerference:STATe) then, the instrument will indicate error -400.

### :FETCh:CINTerference:CIRAtio?

Title: Fetch Carrier to Interference Ratio

Description: Returns the most recent Carrier-to-Interference Ratio measurement result.

Use :FETCh:CINTerference? for complete Carrier-to-Interference Measurement data. If the instrument is sweeping, it will not return until the sweep is complete. If the instrument is not sweeping and the current data is not valid it will return error -230. This could occur if there was a *RST immediately before the :FETCh? or if a measurement parameter was changed without an :INITiate.

If the measurement is not enabled (see [:SENSe]:CHPower:STATe) then, the instrument will indicate error -400.

### :FETCh:CINTerference:CPOWer?

Title: Fetch C/I Carrier Channel Power

Description: Returns the most recent C/I carrier channel power measurement result. Use :FETCh:CINTerference? for complete Carrier-to-Interference Measurement data. If the instrument is sweeping, it will not return until the sweep is complete. If the instrument is not sweeping and the current data is not valid it will return error -230. This could occur if there was a *RST immediately before the :FETCh? or if a measurement parameter was changed without an :INITiate.

If the measurement is not enabled (see [:SENSe]:CHPower:STATe) then, the instrument will indicate error -400.

### :FETCh:CINTerference:IPOWer?

Title: Fetch C/I Interference Channel Power

Description: Returns the most recent C/I interference channel power measurement result. Use :FETCh:CINTerference? for complete Carrier-to-Interference Measurement data. If the instrument is sweeping, it will not return until the sweep is complete. If the instrument is not sweeping and the current data is not valid it will return error -230. This could occur if there was a *RST immediately before the :FETCh? or if a measurement parameter was changed without an :INITiate.

If the measurement is not enabled (see [:SENSe]:CHPower:STATe) then, the instrument will indicate error -400.

### :FETCh:CONStellation:PBCH?

Title: Fetch PBCH Constellation

Description: Get the latest Physical Broadcast Channel (PBCH) constellation hitmap results for the selected beam after waiting for the current measurement to complete. Beam selection is performed using the [:SENSe]:CONStellation:PBCH:BEAM command. The results are in JSON format with the hitmap represented as an array of arrays [I,Q,C] where I = real coordinate, Q = imaginary coordinate, and C = hit count. An example response looks like this, but real responses will likely have many more values than shown here:

{

"qpsk":[[0.673828,0.697266,1],[-0.732422,-0.720703,1],[0.708984,-0.720703,5]],

"carrierId":1

}

In NRAD mode, there will be an additional key-value pair indicating the selected beam number.

"beam":17

### :FETCh:CONStellation:PDSCh?

Title: Fetch Data Channel Constellation

Description: Get the latest Physical Downlink Shared Channel (PDSCH) constellation results after waiting for the current measurement to complete. The results are in JSON format with the hitmap represented as an array of arrays [I,Q,C] where I = real coordinate, Q = imaginary coordinate, and C = hit count. Only constellations that match the selected modulation type (see [:SENSe]:MODulation:TYPE) are returned. An example response looks like this, but real responses will likely have many more values than shown here:

{

"carrierId":1,

"qam64":[[0.779297,-0.427734,1],[0.169922,0.439453,1],[0.169922,-0.181641,1]],

}

**:FETCh:CONTrol?**

Title: Fetch Control Channel Measurement

Description: Returns the Control Channel measurements in json format. This includes Sync, Reference, Broadcast and Control channel measurements. If a measurement is in progress, it blocks until complete.

{
"antennaOneOn":true,
"antennaThreeOn":true,
"antennaTwoOn":false,
"antennaZeroOn":false,
"carrierId":1,
"cellId":91,
"cellIdGroup":30,
"cfi":2,
"correctedCarrierFrequency":-201,
"detectedAntennaCount":1,
"detectedCpType":1,
"frameStart":123456,
"frequencyError":-200,
"gps":{
"coordinates":"37.146865, -121.656103,112m",
"time":"Sat Aug 31 00:44:13 2020 GMT"
},
"pbch":{
"evm":{
"peakEvm":{
"percent":15,
"subcarrierNumber":3,
"symbolNumber":4
},
"rmsEvm":3
},
"power":{
"perReWatts":1.1939880550926317e-09,
"perReDBM":-59.23000017958162,
"totalWatts":9.885530971587295e-10,
"totalDBM":-60.04999998900905
}
},
"pcfich":{

```
"evm":{
"peakEvm":{
"percent":25,
"subcarrierNumber":10,
"symbolNumber":0
},
"rmsEvm":5
},
"power":{
"perReWatts":3.0902954222256074e-09,
"perReDBM":-55.10000001445821,
"totalWatts":9.549926289764696e-10,
"totalDBM":-60.19999980465677
}
},
"pdcch":{
"evm":{
"peakEvm":{
"percent":35,
"subcarrierNumber":56,
"symbolNumber":1
},
"rmsEvm":7
},
"power":{
"perReWatts":1.4125375855655877e-09,
"perReDBM":-58.499999874118416,
"totalWatts":6.095368689784659e-10,
"totalDBM":-62.15000020136438
}
},
"phich":{
"evm":{
"peakEvm":{
"percent":30,
"subcarrierNumber":20,
"symbolNumber":1
},
"rmsEvm":6
},
```

```
"power":{
"perReWatts":1.892343561848973e-09,
"perReDBM":-57.230000130347314,
"totalWatts":7.533555503869141e-10,
"totalDBM":-61.230000076941664
}
},
"ppmFrequencyError":200000000,
"pss":{
"evm":{
"peakEvm":{
"percent":5,
"subcarrierNumber":1,
"symbolNumber":2
},
"rmsEvm":1
},
"power":{
"perReWatts":7.063175644361763e-10,
"perReDBM":-61.509999937447816,
"totalWatts":2.259435859741644e-09,
"totalDBM":-56.459999827929025
}
},
"rs":{
"evm":{
"peakEvm":{
"percent":20,
"subcarrierNumber":4,
"symbolNumber":0
},
"rmsEvm":4
},
"power":{
"perReWatts":7.277797976712457e-09,
"perReDBM":-51.38000004096958,
"totalWatts":1.224616244144273e-08,
"totalDBM":-49.11999984084157
}
},
```

"sectorId":1,
"sss":{
"evm":{
"peakEvm":{
"percent":10,
"subcarrierNumber":2,
"symbolNumber":3
},
"rmsEvm":2
},
"power":{
"perReWatts":1.2941958402734608e-09,
"perReDBM":-58.880000003948076,
"totalWatts":1.0715192502175336e-09,
"totalDBM":-59.700000223000316
}
},
"status":{
"hardwareErrors":{
"ADCOverRange":0,
"DeviceAccess":0,
"EEPROM":0,
"IrregularData":0,
"Limit":0,
"LockLO1":0,
"LockLO2":0,
"NoCal":0,
"Overpower":0,
"SLOLock":0,
"Saturation":0,
"TgLockLo1":0
},
"overallStatusMessage":"statusControlChanMeasDone",
"overallStatusNumber":19
"statusMessage": "statusPciDone",
"statusNumber": 10
},
"timeOffset":1000
"totalNonSharedChannelsPowerWatts":5.260172838461585e-08,
"totalNonSharedChannelsPowerDBM":-42.78999985588293}

}

### :FETCh[:DATA]:ALL? [<ASCii|INTeger|REAL>],[<numeric_value>]

Title: Fetch Trace Data All

Description: This command transfers trace data and trace status data from the instrument to the controller for all traces with display state ON (see Trace Display State commands), as well as measurement data for the currently active measurement (see ":DISPlay:MEASurement") in JSON format.

An optional parameter can be given to set the format data output, the parameter follows the same enumerable value as the Trace Data format setting. When using REAL as the output format, the numeric value field is required. See the document on Trace Data Format (:FORMat[:TRACe][:DATA]) for more info.

The returned response is formatted as a list of packets, each of which has a header detailing its contents. A single value (numeric ascii if ascii format is specified, or uint8 otherwise) precedes the packet list that specifies the number of packets contained in the response. Overall format is as follows:

<IEEE definite length arbitrary block header><Number of packets:uint8><Packet #1><Packet #2>...<Packet #N>

The headers of each packet are either in a comma-separated list if the user requests ASCii format, or in binary otherwise. The individual packets are formatted as follows (value order is preserved between binary and ascii):

<Header Length:uint16><Packet Type:uint8><Trace number:uint8><Packet Data Payload Length:uint32><DataFormat:uint8> <Zero Padding:3 Bytes><Data Payload>

Packet type is represented as: 0 -> Trace Data, 1 -> Trace Status, 2 -> JSON Measurement data

Data Format is represented as: 0 -> ASCII, 1 -> Integer32, 2 -> Real32, 3 -> Real64

The measurement data is in a JSON format with an extra item added under the key "measurementName" indicating the active measurement.

Parameter(s): [<ASCii|INTeger|REAL>],[<numeric_value>]

### :FETCh:DENSity:DATA? <numeric_value>

Title: RTSA density data

Description: In RTSA modes, returns the density data

Parameter(s): <numeric_value>

### :FETCh:EIRPower?

Title: Fetch EIRP

Description: This command returns the current EIRP, Max EIRP, Horizontal EIRP, Vertical and Sum EIRP results in dBm. If the instrument is sweeping, it will not return until the sweep is complete. If the instrument is not sweeping and the current data is not valid it will return error -230. This could occur if there was a *RST immediately before the :FETCh? or if a measurement parameter was changed without an :INITiate. Data is returned as 2 comma-separated values: EIRP power, Max power. If the measurement is not active, the instrument will indicate error -400 and return the string "nan,nan".

**:FETCh:EIRPower:DATA?**

Title: Fetch Current EIRP

Description: This command returns the current EIRP measurement result in dBm.

If the instrument is sweeping, it will not return until the sweep is complete.

If the instrument is not sweeping and the current data is not valid it will return error -230. This could occur if there was a *RST immediately before the :FETCh? or if a measurement parameter was changed without an :INITiate. If the measurement is not active, the instrument will indicate error -400 and return the string "nan".

**:FETCh:EIRPower:MAXimum?**

Title: Fetch MAXimum EIRP

Description: This command returns the maximum EIRP measurement result in dBm.

If the instrument is sweeping, it will not return until the sweep is complete.

If the instrument is not sweeping and the current data is not valid it will return error -230. This could occur if there was a *RST immediately before the :FETCh? or if a measurement parameter was changed without an :INITiate. If the measurement is not active, the instrument will indicate error -400 and return the string "nan".

**:FETCh:FRAMe:POWer?**

Title: Fetch Frame Power Analysis Measurements

Description: Returns the latest Frame Power Analysis measurements in json format. {

"carrierId":1,

"dwPtsPower":-11.686997248037647,

"framePowerDisplay":{

"displayPoints":4001,

"startTime":0,

"stopTime":0

},

"gps":{

"coordinates":"37.146865, -121.656103,112m",

"time":"Sat Aug 31 00:44:13 2020 GMT"

},

"status":{

"hardwareErrors":{

"ADCOverRange":0,

"DeviceAccess":0,

"EEPROM":0,

"IrregularData":0,

"Limit":0,

"LockLO1":0,

"LockLO2":0,

"NoCal":0,

"Overpower":0,

"SLOLock":0,

"Saturation":0,

"TgLockLo1":0

},

"statusMessage":"statusFramePowerMeasDone",

"statusNumber":27

},

"subframePowers":[-11.689188015956093,-12.716933625147382,-62.51848880635962,-62.53143933862779,-61.67019300047982,

-11.670055450451763,-11.670313553247079,-11.664923673930858,-11.665172925879165,-11.664404801000856],

"slotPower":{

"slotPower":
[-80.54065996005235,-81.64280984558984,-89.53351755973559,-92.06537520209018,-87.16220702208979,-86.59842952593326,-90.09767346186983,-90.4888434578913,-87.21263721353337,-87.07058138343614,-85.63481509445133,-87.44050546781867,-85.9671934

8317012,-85.75635347781137,-88.8697573461123,-90.30780137936135,-87.33643988270461,-87.64654104952463,-87.83688674927848,-89.51407913594818],

}, },

"timingError":null,

"guardPeriodPower": nan,

"txTransmitOffPower": nan,

"totalPower":-13.354569314227396,

"traceMeasurementPower":[-7.5062150955200195,-6.351113319396973,...,-11.05777645111082,-10.479146957397464]},

"upPtsPower":-62.51650224671434,

"possibleSsfConfigsList":[ssf0,ssf3]

}

**:FETCh:MIMO:ANTennas?**

Title: Fetch MIMO Antennas

Description: Returns the MIMO Antennas measurements in json format. If a measurement is in progress, it blocks until complete.


{
"antennaOneOn":true,
"antennaOnePower":-83.17574100587828,
"antennaThreeOn":true,
"antennaThreePower":-83.66073582353636,
"antennaTwoOn":false,
"antennaTwoPower":null,
"antennaZeroOn":false,
"antennaZeroPower":null,
"carrierId":1,
"detectedAntennaCount":2,
"gps":{
"coordinates":"nan, nan,nanm",
"time":""
},

"status":{
"hardwareErrors":{
"ADCOverRange":0,
"DeviceAccess":0,
"EEPROM":0,
"IrregularData":0,
"Limit":0,
"LockLO1":0,
"LockLO2":0,
"NoCal":0,
"Overpower":0,
"SLOLock":0,
"Saturation":0,
"TgLockLo1":0
},
"statusMessage":"statusAntDetectionDone",
"statusNumber":14
}
}

### :FETCh:OBWidth<n>?

Title: Fetch Occupied Bandwidth

Description: Returns a different set of measurement information depending on the suffix. The default suffix of 1 will return the most recent occupied bandwidth measurement results: occupied bandwidth, percent of power and dB down. One of either percent of power or dB down is measured and the other is set. That is determined by the value set using [:SENSe]:OBWidth:METHod. If the measurement is not enabled with :SENSe:OBWidth:STATe then, the instrument will indicate error -400 and return the string "nan,nan,nan".

Using suffix 2 will return the most recent channel power, x dB bandwidth, percent bandwidth, and transmit frequency error. The channel power and occupied bandwidth measurements will be calculated regardless of CHP:STAT and OBW:STAT settings.

Using suffix 3 will return the same results as 1, plus the start, center, and end frequencies of the occupied bandwidth.

For all suffixes, if the instrument is sweeping, it will not return until the sweep is complete. If the instrument is not sweeping and the current data is not valid it will return error -230. This could occur if there was a *RST immediately before the :FETCh? or if a measurement parameter was changed without an :INITiate.

OBWidth Suffix
Range: 1-3, Default = 1

### :FETCh:PANalyzer?

Title: Fetch Pulse Analyzer

Description: Return the current Pulse Analyzer measurement data. If the instrument is sweeping, it will not return until the sweep is complete. This command returns pulse waveform measurement duty factor, pulse duration, pulse period, wave average, wave amplitude, pulse center instant, pulse repetition frequency pulse peak, tilt, peak to wave average, pulse overshoot, low (10%) reference level, duration (50%) reference level, high (90%) reference level. It also provide timing measurement for first transition duration start/stop time, second transition duration start/stop time, post/pre transition start/stop time.

An example JSON output is below. Note that the name-value pairs in the response may be in a different order:

{"dutyFactor":"0.5","firstTransitionDuration":"0","firstTransitionDurationRefLevelInstantTime":"0",
"firstTransitionStartTime":"0","firstTransitionStopTime":"0","firstTransitionValid":"true","offTime":"1",
"peakToWaveAverage":"3.0089268140032113","period":"2","postTransitionRegionStartTime":"1",
"postTransitionRegionStopTime":"1","preTransitionRegionStartTime":"0","preTransitionRegionStopTime":"0",
"pulseAverage":"2.0000000000000004","pulseCenterInstant":"0","pulseDuration":"1","pul

seDurationEndTime":"1",
"pulseDurationStartTime":"0","pulsePeak":"2.0000000000000004","pulsePostTransition
Overshoot":"0",
"pulsePostTransitionUndershoot":"0","pulsePreTransitionOvershoot":"0","pulsePreTrans
itionUndershoot":"0",
"pulseReferenceLevelHigh":"-18.30892596786468","pulseReferenceLevelHighState":"2.0
000000000000004",
"pulseReferenceLevelLow":"-26.807733050540033","pulseReferenceLevelLowState":"-33"
,
"pulseReferenceLevelPulseDuration":"-20.7452560658091","pulseRepetitionFrequency":"
5e+8","pulseTransitionValidity":"true",
"secondTransitionDuration":"0","secondTransitionDurationRefLevelInstantTime":"1","se
condTransitionStartTime":"1",
"secondTransitionStopTime":"1","secondTransitionValid":"true","tilt":"0","waveAmplitud
e":"35","waveAverage":"-1.008926814003211"
}

**:FETCh:PCI?**

Title: Fetch PCI Measurements

Description: Returns the PCI measurements. This command returns cell id, group id, corrected carrier frequency, detected beam,

frame start, frequency error, sector id, group id, and time offset(in ns) as output in json format.

If a measurement is in progress, it blocks until complete.

An example is as follows:

{
"carrierId":1,
"cellId": 0,
"cellIdGroup": 0,
"correctedCarrierFrequency": 4499999985.264854,
"detectedBeam": 4,
"frameStart": 233097,
"frequencyError": -14.735145792134368,
"ppmFrequencyError": 0.09346635883256266,
"sectorId": 0,
"status": {
"hardwareErrors": {
"ADCOverRange": 0,
"DeviceAccess": 0,
"EEPROM": 0,
"IrregularData": 0,
"Limit": 0,
"LockLO1": 0,
"LockLO2": 0,
"NoCal": 0,
"Overpower": 0,
"SLOLock": 0,
"Saturation": 0,
"TgLockLo1": 0
},
"statusMessage": "statusPciDone",
"statusNumber": 10
},
"gps":{
"coordinates":"37.14681, -121.65609, 128m",
"time":"Fri Jun 28 19:56:45 2019 GMT"
},

"timeOffset": -634782.9687499882
}

**:FETCh:PDSCh?**

Title: Fetch Data Channel Measurement

Description: Returns the Data Channel measurements in json format.

If a measurement is in progress, it blocks until complete.


{
"activeRbCount":-1,
"carrierId":1,
"gps":{
"coordinates":"37.146865, -121.656103,112m",
"time":"Sat Aug 31 00:44:13 2019 GMT"
},
"ostp":-27.704519932556128,
"percentUtilization":17.430730819702148,
"qam16":{
"peakEvm":{
"percent":13.430730819702148,
"subcarrierNumber":12,
"symbolNumber":5
},
"percentUtilization":0.89,
"rmsEvm":1.7515575885772705
},
"qam256":{
"peakEvm":{
"percent":13.430730819702148,
"subcarrierNumber":12,
"symbolNumber":5
},
"percentUtilization":1.0,
"rmsEvm":1.7515575885772705
},
"qam64":{
"peakEvm":{
"percent":13.430730819702148,
"subcarrierNumber":12,
"symbolNumber":5
},
"percentUtilization":0,
"rmsEvm":1.7515575885772705
},

"qpsk":{
"peakEvm":{
"percent":13.430730819702148,
"subcarrierNumber":12,
"symbolNumber":5
},
"percentUtilization":95,
"rmsEvm":1.7515575885772705
},
"pdschPeakEvm":8.5812377270588576,
"pdschRmsEvm":1.7515575885772705,
"status":{
"hardwareErrors":{
"ADCOverRange":0,
"DeviceAccess":0,
"EEPROM":0,
"IrregularData":0,
"Limit":0,
"LockLO1":0,
"LockLO2":0,
"NoCal":0,
"Overpower":0,
"SLOLock":0,
"Saturation":0,
"TgLockLo1":0
},
"statusMessage":"statusDataChanMeasDone",
"statusNumber":20
}
}

**:FETCh:PEAK?**

Title: Fetch Peak

Description: Returns a pair (frequency in Hz, amplitude) of the peak amplitude in the current sweep. The command does not wait for the sweep to complete. If the trace data is invalid then both numbers in the pair will be NAN and error code -230 will be indicated.

**:FETCh:RRM?**

Title: Fetch RRM Measurements

Description: Returns the Radio Resource Management(RRM) measurements in json format.

RRM measurements are comprised of RSRP, RSRQ, and SINR which are computed using the Cell Reference Signals (CRS).

If a measurement is in progress, it blocks until complete.

**:FETCh:SCANner?**

Title: Fetch Cell Scanner

Description: Returns the cell scanner measurements in a json format. If a measurement is in progress, it blocks until completed. Returned data format varies based on whether the current mode is 5GNR or LTE.

5GNR Mode format:


{
"carrierId":1,
"gps":{
"coordinates":"37.14681, -121.65609, 128m",
"time":"Fri Jun 28 19:56:45 2019 GMT"
},
"status":{
"hardwareErrors":{
"ADCOverRange":0,
"DeviceAccess":0,
"EEPROM":0,
"IrregularData":0,
"Limit":0,
"LockLO1":0,
"LockLO2":0,
"NoCal":0,
"Overpower":0,
"SLOLock":0,
"Saturation":0,
"TgLockLo1":0
},
"statusMessage":"statusCellScannerDone",
"statusNumber":13
},
"values":[
{
"cellId":1007,
"cellIdGroup":335,
"frameStart":1920521,
"sectorId":0,
"timeOffset":-3490194.091145895,
"values":[
{
"beamIndex":0,

"beamPresent":1,
"beamTimeOffset":-3490196.091145895,
"ssRmsEvm":6.488177299499512,
"ssRsrp":-67.18608535715634,
"ssRsrq":-9.913812153567978,
"ssSinr":23.757546218871823
},
...
{
"beamIndex":7,
"beamPresent":1,
"beamTimeOffset":-3490204.091145895,
"ssRmsEvm":6.488177299499512,
"ssRsrp":-84.22420720274197,
"ssRsrq":null,
"ssSinr":null
}
]
}
]
}

LTE Mode format:

{
"carrierId":1,
"gps":{
"coordinates":"nan, nan,nanm","time":""
},
"status":{
"hardwareErrors":{
"ADCOverRange":0,
"DeviceAccess":0,
"EEPROM":0,
"IrregularData":0,
"Limit":0,
"LockLO1":0,
"LockLO2":0,
"NoCal":0,
"Overpower":0,
"SLOLock":0,
"Saturation":0,

```
"TgLockLo1":0
},
"statusMessage":"statusCellScannerDone",
"statusNumber":13
},
"values":[
{
"cellId":2,
"cellIdGroup":0,
"crsRsrp":-68.46522521972656,
"crsRsrq":0.2733601927757263,
"crsSinr":34.74685287475586,
"antPortUsedForMeas":0,
"dominance":null,
"dominantCell":true,
"frameStart":-1,
"frequencyError":227.3341827392578,
"pbch":{
"peakEvm":{
"percent":4.949735641479492,
"subcarrierNumber":-1,
"symbolNumber":-1
},
"rmsEvm":2.2826967239379883
},
"ppmFrequencyError":0.06495262363978795,
"sectorId":2,
"ssRsrp":-68.43709564208984
}
]
}
```

**:FETCh:SEMask?**

Title: Fetch Spectral Emission Mask

Description: This command returns the current Spectral Emission Mask mesurement result. If the instrument is sweeping, it will not return until the sweep is complete. If the instrument is not sweeping and the current data is not valid, it will return an empty JSON object and error -230. This could occur if there was a *RST immediately before the :FETCh? or if a measurement parameter was changed without an :INITiate. Data is returned as ASCII of JSON object that consists of reference power, reference channel power, and an array of segment information. The following is the format of the response: { "referencePwr": <amplitude value>, "referenceChannelPwr": <amplitude value>, "segments": [{"setup": {"rbw": <frequency value>, "points": [{"x": <frequency value>, "y": <amplitude value>}, {"x": <frequency value>, "y": <amplitude value>}]}, "result": {"peakPwr": <amplitude value>, "peakFreq": <amplitude value>, "pass": <true|false>}}, { ... }]

} If the measurement is not active, the instrument will indicate error -400 and return an empty object.

**:FETCh:SSB?**

Title: Fetch SSB Measurements

Description: Returns the SSB measurements.

This command returns the beam numbers with the best EVM and power, followed by Power and EVM results

for PSS, SSS, PBCH, PBCH DM-RS and SS Signal Quality Results in a json format.

If a measurement is in progress, it blocks until complete.

eg:

{
"bestEvmBeam":5,
"bestPowerBeam":0,
"carrierId":1,
"status": {
"hardwareErrors": {
"ADCOverRange": 0,
"DeviceAccess": 0,
"EEPROM": 0,
"IrregularData": 0,
"Limit": 0,
"LockLO1": 0,
"LockLO2": 0,
"NoCal": 0,
"Overpower": 0,
"SLOLock": 0,
"Saturation": 0,
"TgLockLo1": 0
},
"statusMessage": "statusPciDone",
"statusNumber": 10
},
"gps":{
"coordinates":"37.14681, -121.65609, 128m",
"time":"Fri Jun 28 19:56:45 2019 GMT"
},
"values":[
{
"dmRsPbch":{
"peakEvm":{
"percent":11.926020622253418,
"subcarrierNumber":1529,

```
"symbolNumber":3
},
"power":-31.813847928749347,
"rmsEvm":3.813983917236328
},
"pbch":{
"peakEvm":{
"percent":13.430730819702148,
"subcarrierNumber":1638,
"symbolNumber":5
},
"power":-31.8139914869584,
"rmsEvm":3.736370325088501
},
"pss":{
"peakEvm":{
"percent":12.912864685058594,
"subcarrierNumber":1638,
"symbolNumber":2
},
"power":-31.704519932556128,
"rmsEvm":1.7515575885772705
},
"ssRsrp":-31.750750515723816,
"ssRsrq":-10.728772378478801,
"ssSinr":-13.878087163543444,
"sss":{
"peakEvm":{
"percent":11.369827270507812,
"subcarrierNumber":1638,
"symbolNumber":4
},
"power":-31.750750515723816,
"rmsEvm":2.5253000259399414
}
},
{ ... },
{ ... },
{ ... },
{ ... },
```

```
{ ... },
{ ... },
{ ... }
]
}
```

**:FETCh:SYNC:EVM?**

Title: Fetch Sync EVM

Description: Returns the Sync EVM measurements in json format.

If a measurement is in progress, it blocks until complete.


{
"carrierId":1,
"gps":{
"coordinates":"37.146865, -121.656103,112m",
"time":"Sat Aug 31 00:44:13 2019 GMT"
},
"pbch":{
"peakEvm":{
"percent":13.430730819702148,
"subcarrierNumber":12,
"symbolNumber":5
},
"rmsEvm":1.7515575885772705
},
"pss":{
"peakEvm":{
"percent":13.430730819702148,
"subcarrierNumber":12,
"symbolNumber":5
},
"rmsEvm":1.7515575885772705
},
"sss":{
"peakEvm":{
"percent":13.430730819702148,
"subcarrierNumber":12,
"symbolNumber":5
},
"rmsEvm":1.7515575885772705
},
"status":{
"hardwareErrors":{
"ADCOverRange":0,
"DeviceAccess":0,
"EEPROM":0,
"IrregularData":0,

"Limit":0,
"LockLO1":0,
"LockLO2":0,
"NoCal":0,
"Overpower":0,
"SLOLock":0,
"Saturation":0,
"TgLockLo1":0
},
"statusMessage":"statusSyncSigEvmMeasDone",
"statusNumber":18
}
}

### :FETCh:SYNC:POWer?

Title: Fetch Sync Power

Description: Returns the sync power measurements in json format. If a measurement is in progress, it blocks until complete.

{
"carrierId":1,
"gps": {
"coordinates":"37.146865, -121.656103,112m",
"time":"Sat Aug 31 00:44:13 2019 GMT"
},
"pbchPower":-48.73955825818837,
"pssPower":-48.870106550516724,
"sssPower":-48.8758737536726,
"status":{
"hardwareErrors":{
"ADCOverRange":0
},
"statusMessage":"statusSyncandBchPwrDone",
"statusNumber":12
}
}

**:FETCh:TAE?**

Title: Fetch Time Alignment Error

Description: Returns the Time Alignment Error measurements in json format.

If a measurement is in progress, it blocks until complete.


{
"TimeAlignmentErrors":[
{
"antennaPair":{
"alignmentError":2.6679842335397552e-08,"antennaNumbers":"1,2"
}},
{
"antennaPair":{
"alignmentError":-1.845241115461249e-08,"antennaNumbers":"1,3"
}},
{
"antennaPair":{
"alignmentError":-4.5132253490010044e-08,"antennaNumbers":"2,3"
}}
],
"antennaTimeOffsets":{
"antennaOne":1.6969097771379893e-07
"antennaThree":1.8814338886841142e-07,
"antennaTwo":1.4301113537840138e-07,
"antennaZero":null,
},
"carrierId":1,
"gps":{
"coordinates":"37.146865, -121.656103,112m",
"time":"Sat Aug 31 00:44:13 2019 GMT"
},
"status":{
"hardwareErrors":{
"ADCOverRange":0,
"DeviceAccess":0,
"EEPROM":0,
"IrregularData":0,
"Limit":0,
"LockLO1":0,
"LockLO2":0,
"NoCal":0,

"Overpower":0,
"SLOLock":0,
"Saturation":0,
"TgLockLo1":0
},
"statusMessage":"statusAntPortTimeAlignMeasDone",
"statusNumber":22
}
}

### :FETCh:TRACe:STAT?

Title: Fetch Trace Power Statistics

Description: Returns the following value a peak amplitude, minimum amplitude, and average amplitude in the current sweep trace data. If a sweep is in progress, it blocks until complete.

**:FETCh:WCDMa:SYNC?**

Title: Fetch Sync Measurements

Description: Returns the sync measurements. This command returns frequency error stats, carrier id, corrected carrier frequency, frame start, frequency error, gps info, scrambling code, hardware status and time offset(in ns) as output in json format.

If a measurement is in progress, it blocks until complete.

An example is as follows:

{
"FrequencyErrorStats": {
"Count": 185364,
"Deviation": 1.8153972412361203,
"DeviationUnits": "Hz",
"Max": 117.21477065686643,
"Mean": 110.26521356758234,
"Min": 107.69124579804725,
"Units": "Hz"
},
"anritsuHardwareStatus": {
"hardwareErrors": {
"ADCOverRange": 0
},
"statusMessage": "statusSyncDone",
"statusNumber": 10
},
"carrierId": 1,
"correctedCarrierFrequency": 1000000111.0165963,
"frameStart": 39828,
"frequencyError": 111.01659632196692,
"gps": {
"coordinates": "37.146585, -121.656618,128m",
"time": "Tue Jan 1 00:00:00 2024 GMT"
},
"pccpchAbsPower": -48.427171521027155,
"pccpchDeltaPower": -0.3524818420410156,
"pccpchRelPower": -11.313854217529297,
"pcpichEvm": 0.7005375027656555,
"pcpichPower": -48.074689327855744,
"pcpichRelativePower": -10.961372375488281,
"pichAbsPower": -56.04133975526693,
"pichDeltaPower": -7.966650009155273,

"pichRelPower": -18.928022384643555,
"ppmFrequencyError": 0.1110165963219669,
"sccpchAbsPower": -55.5802791084507,
"sccpchDeltaPower": -7.505590438842773,
"sccpchRelPower": -18.466962814331055,
"scramblingCode": 0,
"scramblingCodeGroup": 0,
"timeOffset": 3753108.716643614,
"units": "DBM"
}

**:FORMat[:TRACe][:DATA] <ASCii|INTeger|REAL>,[<numeric_value>]**
**:FORMat[:TRACe][:DATA]?**

Title: Trace Data Format

Description: This command specifies the format in which data is returned in TRAC:DATA queries. The optional numeric parameter is needed for REAL format only. It defines the length of the floating point number in bits. Valid values are 32 and 64. If the optional numeric parameter is omitted, the default length of REAL data is set to 64 bits.

ASCii format returns the data in comma-separated ASCII format. The units are the current measurement units.

INTeger,32 values are signed 32-bit integers in little-endian byte order. This format returns the data in 4-byte blocks. The values are scaled by 1000, so if the current measurement units are dBm the integer values would be mdBm. For example, if the measured result was -12.345 dBm, that value would be sent as -12345.

REAL,32 values are 32-bit floating point numbers conforming to the IEEE 754 standard in little-endian byte order. This format returns the data in 4-byte binary format. The units are the current measurement units.

REAL,64 values are 64-bit floating point numbers conforming to the IEEE 754 standard in little-endian byte order. This format returns the data in 8-byte binary format. The units are the current measurement units.

Parameter(s): <ASCii|INTeger|REAL>,[<numeric_value>]

Query Return: ASC|INT|REAL

Default Value: ASCii

**:FRAMe:SPECtrum:TRACe:DISPlay[:STATe] <0 | 1 | ON | OFF>**
**:FRAMe:SPECtrum:TRACe:DISPlay[:STATe]?**

Title: Trace Display State

Description: The trace visibility state. If it is OFF, the :TRAC:DATA? command will return nan.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**:FRAMe:SPECtrum:TRACe:TYPE**
**<NORMal|MINimum|MAXimum|AVERage|RMAXimum|RMINimum|RAVerage|MATH>**
**:FRAMe:SPECtrum:TRACe:TYPE?**

Title: Trace Type

Description: Specifies how successive sweeps are combined to produce the resulting display value. Setting the TYPE to NORMal will cause the displayed value for a point to be the current measured value for that point. Setting the TYPE to AVERage will cause the displayed value for a point to be the average of the last <integer> measured values where <integer> is set by [:SENSe]:AVERage:COUNt. Setting the TYPE to MAXimum will cause the displayed value for a point is the maximum measured value for that point over sweeps. Setting the TYPE to MINimum will cause the displayed value for a point is the minimum measured value for that point over sweeps.Setting the TYPE to RMAXimum will cause the displayed value for a point to be the maximum of the last <integer> measured values where <integer> is set by [:SENSe]:AVERage:COUNt.Setting the TYPE to RMINimum will cause the displayed value for a point to be the minimum of the last <integer> measured values where <integer> is set by [:SENSe]:AVERage:COUNt.

Parameter(s):

<NORMal | MINimum | MAXimum | AVERage | RMAXimum | RMINimum | RAVerage | MATH>

Query Return: NORM | MIN | MAX | AVER | RMAX | RMIN | RAV | MATH

Default Value: NORMal

**:FRAMe:SPECtrum:TRACe:UPDate[:STATe] <0 | 1 | ON | OFF>**
**:FRAMe:SPECtrum:TRACe:UPDate[:STATe]?**

Title: Trace Update State

Description: The trace update state detemine whether the trace is update in every sweep.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**:FRAMe:TDOMain:TRACe:DISPlay[:STATe] <0 | 1 | ON | OFF>**
**:FRAMe:TDOMain:TRACe:DISPlay[:STATe]?**

Title: Trace Display State

Description: The trace visibility state. If it is OFF, the :TRAC:DATA? command will return nan.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**:FRAMe:TDOMain:TRACe:TYPE**
**<NORMal|MINimum|MAXimum|AVERage|RMAXimum|RMINimum|RAVerage|MATH>**
**:FRAMe:TDOMain:TRACe:TYPE?**

Title: Trace Type

Description: Specifies how successive sweeps are combined to produce the resulting display value. Setting the TYPE to NORMal will cause the displayed value for a point to be the current measured value for that point. Setting the TYPE to AVERage will cause the displayed value for a point to be the average of the last <integer> measured values where <integer> is set by [:SENSe]:AVERage:COUNt. Setting the TYPE to MAXimum will cause the displayed value for a point is the maximum measured value for that point over sweeps. Setting the TYPE to MINimum will cause the displayed value for a point is the minimum measured value for that point over sweeps.Setting the TYPE to RMAXimum will cause the displayed value for a point to be the maximum of the last <integer> measured values where <integer> is set by [:SENSe]:AVERage:COUNt.Setting the TYPE to RMINimum will cause the displayed value for a point to be the minimum of the last <integer> measured values where <integer> is set by [:SENSe]:AVERage:COUNt.

Parameter(s):

<NORMal | MINimum | MAXimum | AVERage | RMAXimum | RMINimum | RAVerage | MATH>

Query Return: NORM | MIN | MAX | AVER | RMAX | RMIN | RAV | MATH

Default Value: NORMal

**:FRAMe:TDOMain:TRACe:UPDate[:STATe] <0 | 1 | ON | OFF>**
**:FRAMe:TDOMain:TRACe:UPDate[:STATe]?**

Title: Trace Update State

Description: The trace update state detemine whether the trace is update in every sweep.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**:INITiate:CONTinuous <0 | 1 | ON | OFF>**
**:INITiate:CONTinuous?**

Title: Sweep Type

Description: Specifies whether the sweep/measurement is triggered continuously. If the value is set to ON or 1, another sweep/measurement is triggered as soon as the current one completes. If continuous is set to OFF or 0, the instrument remains initiated until the current sweep/measurement completes, then enters the 'idle' state and waits for the :INITiate[:IMMediate] command or for :INITiate:CONTinuous ON.

If :INITiate:CONTinuous is changed to ON before the current sweep/measurement completes, a new sweep/measurement will be continuously triggered as soon as the current sweep/measurement completes. If :INITiate[:IMMediate] is received before the current sweep/measurement completes, it will be ignored. Clients must either wait for the current sweep/measurement to complete before triggering a 'single sweep', or :ABORt the sweep/measurement after setting :INITiate:CONTinuous to OFF (which will cause the instrument to immediately enter the idle state where it can accept new triggers).

The default value is ON. That is, sending :INIT:CONT is equivalent to sending :INIT:CONT ON. The query version of the command returns a 1 if the instrument is continuously sweeping/measuring and returns a 0 if the instrument is in single sweep/measurement mode.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**:INITiate[:IMMediate]**

Title: Initiate Single Sweep

Description: Initiates a sweep/measurement. If :INITiate:CONTinuous is set to ON, or if :INITiate:CONTinuous is set to OFF but the current sweep has not completed yet,

this command is ignored. Use this command in combination with :STATus:OPERation?

or *OPC? to synchronize the capture of one complete set of data. When this command

is sent, the "sweep complete" bit of :STATus:OPERation? is set to 0, indicating that the measurement has not completed. The data collection is then triggered. The controlling program can poll :STATus:OPERation? to determine the status. When the "sweep complete" bit is set to 1, data is ready to be retrieved.

In the case of demodulation measurements, only the *OPC? query is applicable for purposes of synchronization; the query will only return a value when both the sweep and the demodulation have completed.

This command is also overlapped, so alternatively, *OPC? can be used to wait for completion of the measurement without polling. When this command is received, the pending operation bit is set. The pending operation will finish once the sweep/measurement is done. Clients can use *OPC? to 'block' until the sweep/measurement is completed.

**:INITiate[:IMMediate]:ALL**

Title: Initiate Average Count Sweep

Description: Initiates sweep until all active traces reach its average count

**:INITiate:SPA:SELFtest?**

Title: Self Test

Description: Perform a self-test and return the results. The response is formatted as a JSON (http://json.org/) array of name,value pair result objects. Some result objects also include a 'status' property that will be either 'pass' or 'fail' based on whether the criteria for that test was met.

For tests of voltages, generally, the test passes if the measured voltage is within 5% or 10% of the expected voltage. A number of absolute minimum and maximum tolerances have been defined and will override this general behavior for results with a nonsymmetric range about the expected value.

**:INSTrument:ACTive:STATe <0 | 1 | ON | OFF>**
**:INSTrument:ACTive:STATe?**

Title: RemoteSpectrumAnalyzerInUse

Description: The "set" state indicates that the instrument is used by someone.

The "unset" state indicates it is not used by anyone

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:INSTrument:REMote:STATe <0 | 1 | ON | OFF>,<string>**
**:INSTrument:REMote:STATe?**

Title: RemoteSpectrumAnalyzerId

Description: This command can be used to indicate the device is being used by someone. The set command takes two parameters:

- a boolean to indicate whether the device is being used

- a string to indicate the identification of someone currently using the device.

When a value of 1 is passed into the boolean parameter (indicating the device being used), the identification string parameter must be non-empty; otherwise, a scpi error will be generated.

When no longer being used, these settings should be cleared with a boolean parameter of zero. The string paramter does not matter because it will automatically be set to the empty string.

The query command returns the current values of the boolean and the string settings. The remote identification string setting is empty only if it has never been set or has been cleared via this command.

Both settings will be reset after a reboot.

Parameter(s): <0 | 1 | ON | OFF>,<string>

### :MEASure:ACPower?

Title: Measure Adjacent Channel Power Ratio

Description: Only available in Spectrum Analyzer mode. ([:SENSe]:MODE SPEC)

Sets the active measurement to adjacent channel power ratio, sets the default measurement parameters, triggers a new measurement and returns the main channel power, lower adjacent, upper adjacent, lower alternate and upper alternate channel power results. It is a combination of the commands :CONFigure:ACPower; :READ:ACPower? For a description of the default adjacent channel power ratio measurement parameters see :CONFigure:ACPower. To make an adjacent channel power ratio measurement with settings other than the default values send: :CONFigure:ACPower

Commands to set desired settings: :READ:ACPower?

Data is returned as 9 comma-separated values: main channel power, absolute lower adjacent channel power, absolute upper adjacent channel power, absolute lower alternate channel power, absolute upper alternate channel power, relative lower adjacent channel power, relative upper adjacent channel power, relative lower alternate channel power, relative upper alternate channel power.

### :MEASure:CHPower?

Title: Measure Channel Power And Density

Description: Only available in Spectrum Analyzer mode. ([:SENSe]:MODE SPEC)

Sets the active measurement to channel power, sets the default measurement parameters, triggers a new measurement and returns the channel power and channel power density results. It is a combination of the commands :CONFigure:CHPower; :READ:CHPower?

For a description of the default channel power measurement parameters see :CONFigure:CHPower. To make a channel power measurement with settings other than the default values send: :CONFigure:CHPower Commands to set desired settings :READ:CHPower? Data is returned as 2 comma-separated values: channel power, channel power density.

### :MEASure:CHPower:CHPower?

Title: Measure Channel Power

Description: Sets the active measurement to channel power, sets the default measurement parameters, triggers a new measurement and returns channel power as the result. It is a combination of the commands :CONFigure:CHPower; :READ:CHPower:CHPower?

To measure both channel power and channel power density use MEASure:CHPower? For a description of the default channel power measurement parameters see :CONFigure:CHPower. To make a channel power measurement with settings other than the default values send: :CONFigure:CHPower, Commands to set desired settings: :READ:CHPower:CHPower?

### :MEASure:CHPower:DENSity?

Title: Measure Channel Power Density

Description: Sets the active measurement to channel power, sets the default measurement parameters, triggers a new measurement and returns channel power density as the result. It is a combination of the commands :CONFigure:CHPower; :READ:CHPower:DENSity?

To measure both channel power and channel power density use MEASure:CHPower? For a description of the default channel power measurement parameters see :CONFigure:CHPower. To make a channel power measurement with settings other than the default values send: :CONFigure:CHPower, Commands to set desired settings: :READ:CHPower:DENSity?

### :MEASure:CINTerference:CANCel

Title: Cancel CI Measurement

Description: Reset the state of the current Carrier-to-Interference Ratio measurement.

### :MEASure:CINTerference:CONTinue

Title: Continue CI Measurement

Description: Advance the state Inactive/CarrierOn/CarrierOff of the current Carrier-to-Interference Ratio measurement.

**:MEASure:IQ:CAPTure [<PPS>]**

Title: StartIQCapture

Description: This set command is used to start the IQ capture measurement.

If IQ:MODE is SINGle, this command will triger a single I/Q block capture. While the capture is in progress the I/Q Capture bit of STATus:OPERation? will be set to 1. Clients can read the captured data with the TRAC:IQ:DATA? query.

If IQ:MODE is STREAM, this command will start streaming capture of I/Q data. The most recently captured block of I/Q data can be read with the TRAC:IQ:DATA? query. In STREAM capture mode, the capture will not complete until aborted. While streaming is going, clients will need to continuously read captured blocks with TRAC:IQ:DATA?.

Regardless of the capture mode, the capture can be aborted. The capture can be aborted with the ABORt command preferably, though most commands which change hardware settings will also abort the capture (clients should assume that any non-query command sent while a capture is in progress will abort the capture). To determine if the capture was aborted, check the output of STATus:OPERation?.

The capture will also be 'paused' if the instrument detects an overpower or overheat condition: in this situation, any pending TRAC:IQ:DATA? query will immediately return #0 and a device-specific error will be added to the SCPI error queue. When the condition is rectified (either by removing the source of the overpower and closing the overpower relay, or waiting for the instrument to cool down), the capture will automatically restart.

Additionally if the instrument detects a change in reference source (either due to a loss/acquisition of GPS or a connection/disconnection of external reference) a device-specific error will be added to the SCPI error queue.

In either capture mode, this command will do nothing if a capture is already in progress.

The device-specific errors this command adds to the SCPI error queue include a description that looks like the following:

Device-specific error;Reference source changed during capture @ Thu Jun 18 17:02:03 2015

Device-specific error;Capture paused due to overheating @ Thu Jun 18 17:02:03 2015

Device-specific error;Capture paused due to RF overpower @ Thu Jun 18 17:02:03 2015

Each description contains the reason for the error and a timestamp when the error occurred.

In products that support SENSe:SWEep:MODE, if the sweep mode (see SENSe:SWEep:MODE) is not currently FFT, this command will set it to FFT prior to starting the capture.

A optional parameter, PPS, can be set if the capture or stream is to start within a ms of the GPS PPS. This can be used to synchronize captures between multiple MS2090As. If there's no GPS, it will be based on an internal PPS counter that might not have an absolute reference.

Parameter(s): [<PPS>]

**:MEASure:IQ:CAPTure:FAIL <OFF|FIRSt|REPeat>**
**:MEASure:IQ:CAPTure:FAIL?**

Title: Capture on Failure

Description: Sets or queries whether the instrument will automatically save an IQ capture when losing sync.

When the instrument goes from having good sync to no sync, the IQ data for the failed measurement is saved to the user partition.

"FIRSt" causes a single capture to be performed and then transition to the OFF" state. REPeat" causes saves the IQ data on every loss of sync.

"FIRSt" is recommended to avoid accidentally filling up the user partition.

This command is only available when in 5GNR Analyzer mode ([:SENSe]:MODE NRADio).

Parameter(s): <OFF | FIRSt | REPeat>

Query Return: OFF | FIRS | REP

Default Value: OFF

**:MMEMory:COEvent:CLEarall**

Title: Clear All IQ Capture On Event

Description: Turns off all IQ Capture on event types that are active.

**:MMEMory:COEvent:TIMer:INTerval <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**:MMEMory:COEvent:TIMer:INTerval?**

Title: IQ Capture Timer Interval

Description: This command sets or queries the time interval between IQ captures in miliseconds.

Used for Continuous Capture mode only - does not affect Single Capture or Streaming modes.

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 0 ms

Default Unit: ms

Range: 0 ms to 216000000 ms

### :MMEMory:LOAD:LIMit <string>,<string>,<string>

Title: Load Limit From File

Description: This command loads limit data from the specified limit file.

The file location is resolved using the MSUS and file path parameters. Parameters:

1. Label: Used to specify which limit to load. Currently, the label only supports loading all limits. Send an empty string or "ALL" to load all limits.

2. File path: The file path (including file name) relative to the MSUS root directory. File path is case sensitive. File extension is optional.

3. MSUS: Device to load the file from. See MMEMory:CATalog:MSUSs? for information on obtaining the list of available devices. MSUS parameter is case sensitive.

For example:

MMEMory:LOAD:LIMit "ALL","/example_limit_file", "USB1-1"

The model of the device that the limit was saved in must match the model of the the device to load in; otherwise, the load will be rejected.

Parameter(s): <string>,<string>,<string>

### :MMEMory:LOAD:RAM <numeric_value>,<numeric_value>,<string>

Title: Load RAM

Description: Store data to RAM from a file. This command takes three parameters

1. data block size: that needs to be stored in bytes (must be a word aligned)

2. offset: in bytes from start of RAM address to store data at

3. filename: name of the file without the full path. Contents of this file will be copied over to RAM

Parameter(s): <numeric_value>,<numeric_value>,<string>

### :MMEMory:LOAD:RECord <string>,<string>

Title: Load Playback File

Description: This command loads a file made with Trace Recording for playback.

1. File path: The file path (including file name) relative to the MSUS root directory. File extension is optional.

2. MSUS: Device to recall the file from. See MMEMory:CATalog:MSUSs? for information on obtaining the list of available devices.

Parameter(s): <string>,<string>

**:MMEMory:LOAD:SEMask:CUSTom <string>,<string>**

Title: Load Custom SEM From File

Description: This command loads sem mask data from the specified csv file.

The file location is resolved using the MSUS and file path parameters. Parameters:

2. File path: The file path (including file name) relative to the MSUS root directory. File path is case sensitive. File extension is optional.

3. MSUS: Device to load the file from. See MMEMory:CATalog:MSUSs? for information on obtaining the list of available devices. MSUS parameter is case sensitive.

For example:

MMEMory:LOAD:SEMask:CUSTom "/example_mask.csv", "USB1-1"

Parameter(s): <string>,<string>

**:MMEMory:STOEvent:CLEarall**

Title: Clear All Save On Event

Description: Turns off all save on event types that are active.

**:MMEMory:STOEvent:EOSWeep:MODE <CONTinuous|SINGle>**
**:MMEMory:STOEvent:EOSWeep:MODE?**

Title: End of Sweep Save On Event Mode

Description: Specifies the stop mode of the end of sweep save on event system. Setting the value to CONTinuous will cause the instrument to keep saving traces at the completion of every valid sweep. Setting the value to SINGle will trigger a save trace on the next complete valid sweep and then automatically turn the end of sweep save on event feature OFF.

Parameter(s): <CONTinuous|SINGle>

Query Return: CONT|SING

Default Value: CONTinuous

**:MMEMory:STOEvent:EOSWeep[:STATe] <0 | 1 | ON | OFF>**
**:MMEMory:STOEvent:EOSWeep[:STATe]?**

Title: End of Sweep Save on Event State

Description: Turn the end of sweep save on event ON or OFF. Turning the feature on will cause the instrument to automatically save a trace whenever a sweep completes.

A common cause of the command failure is not having enough space available on the storage location. Use :MMEMory:CATalog:DIRectory query command to retrieve the total space available on the storage location.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:MMEMory:STOEvent:LIMit:INTerval <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**:MMEMory:STOEvent:LIMit:INTerval?**

Title: Limit Save On Event Interval

Description: Sets the time interval to keep saving after a limit line failure has occured when :MMEMory:STOEvent:LIMit:MODE is set to INTerval.

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 60000 ms

Default Unit: ms

Range: 0 ms to 216000000 ms

**:MMEMory:STOEvent:LIMit:MODE <CONTinuous|SINGle|INTerval>**
**:MMEMory:STOEvent:LIMit:MODE?**

Title: End of Sweep Save On Event Mode

Description: Specifies the stop mode of the limit save on event system. Setting the value to CONTinuous will cause the instrument to keep saving traces at every limit line failure. Setting the value to SINGle will trigger a save trace on the next limit line failure and then automatically turn the end of sweep save on event feature OFF automatically. Setting the value to INTerval will trigger a save on the next limit line failure and contiue to save at every end of sweep until the set time interval has expired.

Parameter(s): <CONTinuous | SINGle | INTerval>

Query Return: CONT | SING | INT

Default Value: CONTinuous

**:MMEMory:STOEvent:LIMit:PTRigger[:STATe] <0 | 1 | ON | OFF>**
**:MMEMory:STOEvent:LIMit:PTRigger[:STATe]?**

Title: Limit Pretrigger Save on Event State

Description: Turn the limit pre-trigger save on event ON or OFF. Turning the feature ON will cause the instrument to automatically save a trace captured prior to the limit line failure trace. The :MMEMory:STOEvent:LIMit[:STATe] and :INITiate:CONTinuous needs to be turned ON for the pre-trigger feature to apply.

A common cause of the command failure is not having enough space available on the storage location. Use :MMEMory:CATalog:DIRectory query command to retrieve the total space available on the storage location.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:MMEMory:STOEvent:LIMit[:STATe] <0 | 1 | ON | OFF>**
**:MMEMory:STOEvent:LIMit[:STATe]?**

Title: Limit Save on Event State

Description: Turn the limit save on event ON or OFF. Turning the feature on will cause the instrument to automatically save a trace whenever a limit line failure occurs.

A common cause of the command failure is not having enough space available on the storage location. Use :MMEMory:CATalog:DIRectory query command to retrieve the total space available on the storage location.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:MMEMory:STOEvent:TIMer:INTerval <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**:MMEMory:STOEvent:TIMer:INTerval?**

Title: Timer Save On Event Interval

Description: Sets the time interval to save when :MMEMory:STOEvent:TIMer[:STATe] is enabled. The queried value is always in milliseconds.

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 240000 ms

Default Unit: ms

Range: 0 ms to 3600000 ms

**:MMEMory:STOEvent:TIMer[:STATe] <0 | 1 | ON | OFF>**
**:MMEMory:STOEvent:TIMer[:STATe]?**

Title: Limit Save on Event State

Description: Turn the timer save on event ON or OFF. Turning the feature on will cause the instrument to automatically save a trace at the interval specified by :MMEMory:STOEvent:LIMit:INTerval, if space is available. Available memory may be checked using :MMEMory:CATalog:DIRectory query command.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:MMEMory:STORe:CAPTure**

Title: Save IQ Capture

Description: This command performs saving the IQ Capture data to file.

**:MMEMory:STORe:CAPTure:MODE <AUTomatic|MANual>**
**:MMEMory:STORe:CAPTure:MODE?**

Title: IQ Capture Saving Mode

Description: This command sets or queries the saving mode for IQ Capture.

AUTomatic saves every capture to file.

MANual saves the latest capture to file when user send :MMEMory:STORe:CAPTure command.

Parameter(s): <AUTomatic|MANual>

Query Return: AUT|MAN

Default Value: AUTomatic

**:MMEMory:STORe:CAPTure:MSUS <string>**
**:MMEMory:STORe:CAPTure:MSUS?**

Title: IQ Storage Device

Description: This command sets or queries the storage device on which to store IQ capture files. Use the :MMEMory:CATalog:MSUSs? query to retrieve a list of valid parameters for this setting.

Parameter(s): <string>

Default Value: Internal

**:MMEMory:STORe:CAPTure:PATH <string>**
**:MMEMory:STORe:CAPTure:PATH?**

Title: IQ File Location

Description: This command sets or queries the relative path to the directory in which IQ capture files will be saved. The path is relative to the selected IQ storage device specified by the IQ Storage Device setting (MMEM:STOR:CAPT:MSUS).

Parameter(s): <string>

Default Value: Captures

**:MMEMory:STORe:CAPTure:PREFix <string>**
**:MMEMory:STORe:CAPTure:PREFix?**

Title: IQ File Name Prefix

Description: This command sets or queries the file name prefix for IQ Capture or IQ Streaming.

Parameter(s): <string>

Default Value: Capture

**:MMEMory:STORe:CSCanner:CSV:MSUS <string>**
**:MMEMory:STORe:CSCanner:CSV:MSUS?**

Title: Channel Scanner Csv Logging Storage Device

Description: This command sets or queries the storage device where Channel Scanner log files will be saved. Use the :MMEMory:CATalog:MSUSs? query to retrieve a list of valid parameters for this setting.

Parameter(s): <string>

Default Value: Internal

**:MMEMory:STORe:CSCanner:CSV:PATH <string>**
**:MMEMory:STORe:CSCanner:CSV:PATH?**

Title: Channel Scanner CSV Logging Location

Description: This command sets or queries the relative path to the directory wheres Channel Scanner result files will be saved. The log location is only used when Channel Scanner logging is enabled. The path is relative to the selected Channel Scanner storage device specified by the Device setting (:MMEMory:STORe:CSCanner:CSV:MSUS).

Parameter(s): <string>

Default Value: ChannelScannerCsv

**:MMEMory:STORe:LIMit <string>,<string>,<string>**

Title: Store Limit To File

Description: This command stores the limit data into the specified file location. The file location is resolved using the MSUS and file path parameters. Intermediate folders specified by the file path parameter are automatically created. The output file extension is .lim.

The command automatically overwrites the specified file if present. Parameters:

1. Label: Used to specify which limit to store. Currently, the label only supports storing all limits. Send an empty string or "ALL" to store all limits.

2. File path: The file path (including file name) relative to the MSUS root directory. File path is case sensitive. File extension is optional.

3. MSUS: Device to store the file to. See MMEMory:CATalog:MSUSs? for information on obtaining the list of available devices. MSUS parameter is case sensitive.

For example:

MMEMory:STORe:LIMit "ALL","/example_limit_file", "USB1-1"

A common cause of the command failure is not having enough space available on the storage location. Use :MMEMory:CATalog:DIRectory query command to retrieve the total space available on the storage location.

Parameter(s): <string>,<string>,<string>

**:MMEMory:STORe:RAM <numeric_value>,<numeric_value>,<string>**

Title: Store RAM

Description: Load data from RAM to a file. This command takes three parameters

1. data block size: that needs to be loaded in bytes (must be a word aligned)

2. offset: in bytes from start of RAM address to load from

3. filename: name of the file without the full path. This is the destination file where the contents of RAM are copied over In NRAD and LTE modes, this command saves a IQ capture in 32 bit floating point format

matching the settings. The parameters are ignored (but still need to be passed) and values

are used are based on the capture bandwith. The capture is 60 ms long to cover multiple frames.

This is the command used by the capture buttons in the NRAD and LTE modes in the file menu.

Parameter(s): <numeric_value>,<numeric_value>,<string>

**:MMEMory:STORe:RECord:CSV:DISCard**

Title: Discard Trace Recording CSV File

Description: This command discards the last file written for Trace Recording in CSV.

**:MMEMory:STORe:RECord:CSV:MSUS <string>**
**:MMEMory:STORe:RECord:CSV:MSUS?**

Title: Trace Recording CSV Storage Device

Description: This command sets or queries the MSUS device that Trace Recordings in CSV will be stored in.

Parameter(s): <string>

Default Value: Internal

**:MMEMory:STORe:RECord:CSV:PATH <string>**
**:MMEMory:STORe:RECord:CSV:PATH?**

Title: Trace Recording CSV Storage Location

Description: This command sets or queries the directory that Trace Recordings in CSV will be stored in.

Parameter(s): <string>

Default Value: PlaybackCsv

**:MMEMory:STORe:RECord:CSV:PREFix <string>**
**:MMEMory:STORe:RECord:CSV:PREFix?**

Title: Trace Recording CSV File Name Prefix

Description: This command sets or queries the file name prefix for Trace Recording in CSV.

Parameter(s): <string>

Default Value: Traces

**:MMEMory:STORe:RECord:MSUS <string>**
**:MMEMory:STORe:RECord:MSUS?**

Title: Trace Recording Storage Device

Description: This command sets or queries the MSUS device recorded traces will be stored in.

Parameter(s): <string>

Default Value: Internal

**:MMEMory:STORe:RECord:PATH <string>**
**:MMEMory:STORe:RECord:PATH?**

Title: Trace Recording Storage Location

Description: This command sets or queries the path to the directory recorded traces will be stored in.

Parameter(s): <string>

Default Value: Playback

**:MMEMory:STORe:RECord:PREFix <string>**
**:MMEMory:STORe:RECord:PREFix?**

Title: Trace Recording File Name Prefix

Description: This command sets or queries the file name prefix for Trace Recording.

Parameter(s): <string>

Default Value: RecordedSession

**:MMEMory:STORe:STREam:MSUS <string>**
**:MMEMory:STORe:STREam:MSUS?**

Title: IQ Stream USB Storage

Description: This command sets or queries the USB storage device on which to store IQ stream files. Use the :MMEMory:CATalog:MSUSs? query to retrieve a list of valid USB device for this setting.

Parameter(s): <string>

Default Value: None

**:MMEMory:STORe:STREam:PATH <string>**
**:MMEMory:STORe:STREam:PATH?**

Title: IQ Stream File Location

Description: This command sets or queries the relative path to the directory in which IQ stream files will be saved. The path is relative to the USB storage device specified by the IQ Stream USB Storage setting (MMEM:STOR:STRE:MSUS).

Parameter(s): <string>

Default Value: Streams

**:MMEMory:STORe:TRACe:CSV <string>,<string>,<string>**

Title: Store Trace in CSV

Description: This command saves the current trace to CSV file in specified file location. The file location is resolved using the MSUS and file path parameters. These parameters are case sensitive.

Parameters:

1. Label: Used to specify which trace to save. Currently, the label only supports saving all traces. Send an empty string or "ALL" to store all traces.

2. File path: The file path (including file name) relative to the MSUS root directory. File extension should not be specified.

3. MSUS: Device to save the file on. See MMEMory:CATalog:MSUSs? for information on obtaining the list of available devices.

A common cause of the command failure is not having enough space available on MSUS to save trace. Use :MMEMory:CATalog:DIRectory query command to retrieve the total space available on MSUS.

Parameter(s): <string>,<string>,<string>

**:OUTPut:IF:FILTer <K750|M7P5|M14|M30|M100|K40|K800|M5P5|M20|M40>**
**:OUTPut:IF:FILTer?**

Title: IF Filter Select

Description: If the following criteria is met the IF filter can be manually selected: Option 0089 (IF Output) is enabled/Zero Span is selected/IF output is ON. Not available on the MS2080A or MS2070A. Parameter Filter Bandwidth

K750 750kHz

M7P5 7.5MHz

M14 14MHz

M30 32MHz

M100 100MHz

**The 100MHz filter selection is only available with options 0103 and 0104

Parameter(s): <K750|M7P5|M14|M30|M100|K40|K800|M5P5|M20|M40>

Query Return: K750|M7P5|M14|M30|M100|K40|K800|M5P5|M20|M40

Default Value: M100

**:OUTPut:IF:STATe <0 | 1 | ON | OFF>**
**:OUTPut:IF:STATe?**

Title: IF Output State

Description: Toggles the analog IF output on/off. Not available on the MS2080A or MS2070A.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:READ:ACPower?**

Title: Read Adjacent Channel Power Ratio

Description: Triggers a new adjacent channel power ratio measurement and returns the results: main channel power, lower adjacent and upper adjacent channel power. It is the equivalent of the command sequence :ABORT; :INITiate; :FETCh:ACPower?

The channel power measurement must be the active measurement (specified by the command :CONFigure:ACPower).The instrument should be in the single sweep mode.

The current measurement can be queried using the command :CONFigure? Data is returned as 9 comma-separated values: main channel power, absolute lower adjacent channel power, absolute upper adjacent channel power, absolute lower alternate channel power, absolute upper alternate channel power, relative lower adjacent channel power, relative upper adjacent channel power, relative lower alternate channel power, relative upper alternate channel power.

If the measurement is not active, the instrument will indicate error -400 and return the string "nan,nan,nan,nan,nan". If :INITiate command fails or if :INITiate:CONTinuous is set to ON(continuous sweep) it returns a string "nan,nan,nan,nan,nan,nan,nan,nan,nan"

**:READ:CHPower?**

Title: Read Channel Power And Density

Description: Triggers a new channel power measurement and returns the results: channel power and channel power density. It is the equivalent of the command sequence: ":ABORT; :INITiate; :FETCh:CHPower?"

The channel power measurement must be the active measurement (specified by the command :CONFigure:CHPower).The instrument should be in the single sweep mode. Data is returned as 2 comma-separated values: channel power, channel power density.

If the measurement is not active, the instrument will indicate error -400 and return the string "nan,nan". If :INITiate command fails, or if :INITiate:CONTinuous is set to ON(continuous sweep), it returns a string "nan,nan".

**:READ:CHPower:CHPower?**

Title: Read Channel Power

Description: Triggers a new channel power measurement and returns the channel power result: channel power. It is the equivalent of the command sequence ":ABORT; :INITiate; :FETCh:CHPower:CHPower?"

The channel power measurement must be the active measurement (specified by the command :CONFigure:CHPower).The instrument should be in the single sweep mode. Data returned is: channel power.

If the measurement is not active, the instrument will indicate error -400 and return the string "nan,nan". If :INITiate command fails, or if :INITiate:CONTinuous is set to ON(continuous sweep), it returns a string "nan,nan".

**:READ:CHPower:DENSity?**

Title: Read Channel Power Density

Description: Triggers a new channel power measurement and returns the results: channel power density. It is the equivalent of the command sequence :ABORT; :INITiate; :FETCh:CHPower:DENSity? The channel power measurement must be the active measurement (specified by the command :CONFigure:CHPower).The instrument should be in the single sweep mode. Data returned is: channel power density.

If the measurement is not active, the instrument will indicate error -400 and return the string "nan,nan". If :INITiate command fails, or if :INITiate:CONTinuous is set to ON(continuous sweep), it returns a string "nan,nan"

**:READ:EIRPower?**

Title: Read EIRP

Description: Triggers a new EIRP measurement and returns the current EIRP, Max EIRP, Horizontal EIRP, Vertical and Sum EIRP results in dBm. It is a combination of the commands :ABORT; :INITiate; :FETCh:EIRPower?

The EIRP measurement must be the active measurement (specified by the command :CONFigure:EIRPower). The instrument should be in the single sweep mode. Data is returned as 2 comma-separated values: EIRP, max EIRP. If the measurement is not active, the instrument will indicate error -400 and return the string "nan,nan". If :INITiate command fails or if :INITiate:CONTinuous is set to ON(continuous sweep), it returns a string "nan,nan".

**:READ:EIRPower:DATA?**

Title: Read Current EIRP

Description: Triggers a new EIRP measurement and returns the current EIRP result in dBm. It is a combination of the commands :ABORT; :INITiate; :FETCh:EIRPower:DATA?

The EIRP measurement must be the active measurement (specified by the command :CONFigure:EIRPower).The instrument should be in the single sweep mode. If the measurement is not active, the instrument will indicate error -400 and return the string "nan". If :INITiate command fails or if :INITiate:CONTinuous is set to ON(continuous sweep),it returns a string "nan".

**:READ:EIRPower:MAXimum?**

Title: Read MAXimum EIRP

Description: Triggers a new EIRP measurement and returns the maximum EIRP results in dBm. It is a combination of the commands :ABORT; :INITiate; :FETCh:EIRPower:MAX?

The EIRP measurement must be the active measurement (specified by the command :CONFigure:EIRPower). The instrument should be in the single sweep mode. If the measurement is not active, the instrument will indicate error -400 and return the string "nan". If :INITiate command fails or if :INITiate:CONTinuous is set to ON(continuous sweep),it returns a string "nan".

**:READ:SEMask?**

Title: Read Spectral Emission Mask

Description: Triggers a new Spectral Emission Mask measurement and returns the current Spectral Emission Mask mesurement result. It is a combination of the commands :ABORT; :INITiate; :FETCh:SEMask? The Spectracl Emission Mask measurement must be the active measurement (specified by the command :CONFigure:SEMask). Data is returned as ASCII of JSON object. Please refer to :FETCh:SEMask command for sample response.

If the measurement is not active, the instrument will indicate error -400 and return an empty object.

**[:SENSe]:ACPower:BANDwidth|BWIDth:ADJacent <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe]:ACPower:BANDwidth|BWIDth:ADJacent?**

Title: Adjacent CHPWR Adjacent Integration Bandwidth

Description: Sets the adjacent channel bandwidth for adjacent channel power measurement. Integration bandwidth must be less than or equal to span

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

Query Return: Numeric (Hz)

Default Value: 10350000 Hz

Default Unit: Hz

Range: 10 Hz to 400000000000 Hz

**[:SENSe]:ACPower:BANDwidth|BWIDth:ALTernate <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe]:ACPower:BANDwidth|BWIDth:ALTernate?**

Title: Adjacent CHPWR Alternate Integration Bandwidth

Description: Sets the alternate channel bandwidth for adjacent channel power measurement. Integration bandwidth must be less than or equal to span

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

Query Return: Numeric (Hz)

Default Value: 10350000 Hz

Default Unit: Hz

Range: 10 Hz to 400000000000 Hz

**[:SENSe]:ACPower:BANDwidth|BWIDth:MAIN <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe]:ACPower:BANDwidth|BWIDth:MAIN?**

Title: Adjacent CHPWR Main Integration Bandwidth

Description: Sets the main channel bandwidth for adjacent channel power measurement. Integration bandwidth must be less than or equal to span

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

Query Return: Numeric (Hz)

Default Value: 10350000 Hz

Default Unit: Hz

Range: 10 Hz to 400000000000 Hz

**[:SENSe]:ACPower:BANDwidth|BWIDth:SPACing <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe]:ACPower:BANDwidth|BWIDth:SPACing?**

Title: Adjacent Channel Power Spacing

Description: Controls the channel spacing which is the distance from the modulated carrier signal to the

upper or lower Adjacent Channel.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

Query Return: Numeric (Hz)

Default Value: 10350000 Hz

Default Unit: Hz

Range: 10 Hz to 400000000000 Hz

**[:SENSe]:ACPower:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:ACPower:STATe?**

Title: Adjacent Channel Power State

Description: Sets the state of the adjacent channel power ratio measurement, ON or OFF. When using :CONFigure:ACPower, the state is automatically set to ON

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:AMFM:AUDio:RECord**

Title: AMFM Start Recording

Description: This command starts the recording of the current audio output for the specified duration to a .wav file.

**[:SENSe]:AMFM:AUDio:RECord:DURation <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**[:SENSe]:AMFM:AUDio:RECord:DURation?**

Title: AMFM Recording time

Description: This command sets and queries the duration of the audio captures in the AMFM measurement.

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 10000 ms

Default Unit: ms

Range: 1000 ms to 100000000 ms

**[:SENSe]:AMFM:AUDio:RESults?**

Title: AMFM Audio Measurement Results

Description: This command gets the latest AMFM audio measurement results in an unsupported binary format

**[:SENSe]:AMFM:AUDio:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:AMFM:AUDio:STATe?**

Title: AMFM Audio State

Description: This command sets and queries whether or not the audio is on in the AMFM measurement.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:AMFM:DEMod:BWIDth?**

Title: AMFM Demod Bandwidth

Description: This command queries the currently-used bandwidth for the AMFM measurement.

Query Return: Numeric (Hz)

Default Value: 3900 Hz

Default Unit: Hz

Range: 1000 Hz to 40000000 Hz

**[:SENSe]:AMFM:DEMod:FREQuency <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe]:AMFM:DEMod:FREQuency?**

Title: AMFM Demodulation Frequency

Description: This command sets and queries the current demodulation frequency of the AMFM meausurement.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

Query Return: Numeric (Hz)

Default Value: 3000000000 Hz

Default Unit: Hz

Range: -99999999995 Hz to 299999999995 Hz

**[:SENSe]:AMFM:DEMod:MARKer:SELect <numeric_value>**
**[:SENSe]:AMFM:DEMod:MARKer:SELect?**

Title: AMFM Marker Tracking Selected Marker

Description: This command sets and queries which marker the demod frequency will track if [:SENSe]:AMFM:DEMod:MARKer is on.

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 1

Range: 1 to 12

**[:SENSe]:AMFM:DEMod:MARKer:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:AMFM:DEMod:MARKer:STATe?**

Title: AMFM Marker Tracking State

Description: This command sets and queries whether or not the demod frequency will track a selected marker. See [:SENSe]:AMFM:DEMod:MARKer:SELect

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:AMFM:DEMod:TYPE**
**<AM|USBand|LSBand|FMWideband|FMNarrowband|FM125|FM625>**
**[:SENSe]:AMFM:DEMod:TYPE?**

Title: AMFM Demodulation Type

Description: This command sets and queries the current demodulation type of the AMFM meausurement.

Parameter(s): <AM | USBand | LSBand | FMWideband | FMNarrowband | FM125 | FM625>

Query Return: AM | USB | LSB | FMW | FMN | FM125 | FM625

Default Value: AM

**[:SENSe]:AMFM:SQUelch <numeric_value> {<amplitude_units>}**
**[:SENSe]:AMFM:SQUelch?**

Title: AMFM Squelch Level

Description: This command sets and queries the squelch level, which cuts off audio from signals below its level.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {<amplitude_units>}

Query Return: Numeric (<amplitude_units>)

Default Value: -120 dBm

Default Unit: dBm

Range: -120 dBm to 30 dBm

**[:SENSe]:AMFM:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:AMFM:STATe?**

Title: AMFM Measurement State

Description: This command sets and queries the state of the AMFM meausurement.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:ARFChannel <numeric_value>**
**[:SENSe]:ARFChannel?**

Title: Absolute RF Channel Number

Description: Sets or queries the Absolute RF Channel Number for the current demod measurement. Accepted values vary based on the current operating band. Inputting invalid values will set the closest valid value possible for the given circumstances.

When the operating band is not being used ([:SENSe]:OBANd is INValid), this value will not be settable. Attempts to set this when the system is in this state will return the error -221.

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 0

Range: 0 to 3279165

**[:SENSe]:AVERage:COUNt <numeric_value>**
**[:SENSe]:AVERage:COUNt?**

Title: Sense Average Count

Description: Sets the effective number of averages and the length of the rolling min and max hold. Due to memory limitations the actual length of buffers for rolling min and max hold are limited to 2,000,000 / number of trace points.

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 10

Range: 2 to 1000

**[:SENSe]:AVERage:TYPE**
**<NORMal|MINimum|MAXimum|AVERage|RMAXimum|RMINimum|RAVerage|MATH>**
**[:SENSe]:AVERage:TYPE?**

Title: Sense Average Type

Description: Specifies how successive traces are combined to produce the resulting display value. Only applies to trace 1. Setting the TYPE to NORMal will cause the displayed value for a point to be the current measured value for that point. Setting the TYPE to AVERage will cause the displayed value for a point to be the average of the last <integer> measured values where <integer> is set by [:SENSe]:AVERage:COUNt. Setting the TYPE to MAXimum will cause the displayed value for a point is the maximum measured value for that point over sweeps. Setting the TYPE to MINimum will cause the displayed value for a point is the minimum measured value for that point over sweeps.Setting the TYPE to RMAXimum will cause the displayed value for a point to be the maximum of the last <integer> measured values where <integer> is set by [:SENSe]:AVERage:COUNt.Setting the TYPE to RMINimum will cause the displayed value for a point to be the minimum of the last <integer> measured values where <integer> is set by [:SENSe]:AVERage:COUNt.

Parameter(s):

<NORMal | MINimum | MAXimum | AVERage | RMAXimum | RMINimum | RAVerage | MATH>

Query Return: NORM | MIN | MAX | AVER | RMAX | RMIN | RAV | MATH

Default Value: NORMal

**[:SENSe]:BANDwidth|BWIDth[:RESolution] <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe]:BANDwidth|BWIDth[:RESolution]?**

Title: RBW

Description: Sets the resolution bandwidth.

Note that using this command turns the automatic resolution bandwidth setting OFF.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

Query Return: Numeric (Hz)

Default Value: 3000000 Hz

Default Unit: Hz

Range: 1 Hz to 50000000 Hz

**[:SENSe]:BANDwidth|BWIDth[:RESolution] <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe]:BANDwidth|BWIDth[:RESolution]?**

Title: RBW Zero Span

Description: Sets the resolution bandwidth.

Note that using this command turns the automatic resolution bandwidth setting OFF. In Zero Span, the range will change to allow a mininum of 5 KHz to the maximum of 20 MHz.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

Query Return: Numeric (Hz)

Default Value: 3000000 Hz

Default Unit: Hz

Range: 5000 Hz to 20000000 Hz

**[:SENSe]:BANDwidth|BWIDth[:RESolution]:RATio <numeric_value>**
**[:SENSe]:BANDwidth|BWIDth[:RESolution]:RATio?**

Title: RBW Span Ratio

Description: Sets the ratio of the resolution bandwidth to the span for use when the resolution bandwidth to span coupling is enabled. Note that the front panel interface sets the inverse ratio: the span to the resolution bandwidth.

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 0.01

Range: 1e-05 to 1

**[:SENSe]:BANDwidth|BWIDth:VIDeo <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe]:BANDwidth|BWIDth:VIDeo?**

Title: VBW

Description: Sets the video bandwidth.

Note that using this command turns the automatic video bandwidth setting OFF.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

Query Return: Numeric (Hz)

Default Value: 1000000 Hz

Default Unit: Hz

Range: 1 Hz to 50000000 Hz

**[:SENSe]:BANDwidth|BWIDth:VIDeo <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe]:BANDwidth|BWIDth:VIDeo?**

Title: VBW Zero Span

Description: Sets the video bandwidth.

Note that using this command turns the automatic video bandwidth setting OFF. In Zero Span, the range will change to allow a mininum of 5 KHz to the maximum of 20 MHz.

The Default value will also change to 3 MHz.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

Query Return: Numeric (Hz)

Default Value: 3000000 Hz

Default Unit: Hz

Range: 5000 Hz to 20000000 Hz

**[:SENSe]:BANDwidth|BWIDth:VIDeo:RATio <numeric_value>**
**[:SENSe]:BANDwidth|BWIDth:VIDeo:RATio?**

Title: VBW RBW Ratio

Description: Sets the ratio of the video bandwidth to the resolution bandwidth for use when the video to resolution bandwidth coupling is enabled. Note that the front panel interface sets the inverse ratio: the resolution bandwidth to the video bandwidth which is an integer, in other words, if you send 0.35, the display will show 2 not 2.857

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 0.333333333333333

Range: 1e-05 to 1

**[:SENSe]:BANDwidth|BWIDth:VIDeo:TYPE <LINear|LOGarithmic>**
**[:SENSe]:BANDwidth|BWIDth:VIDeo:TYPE?**

Title: VBW Averaging

Description: Changes the VBW/Average type.

Parameter(s): <LINear|LOGarithmic>

Query Return: LIN|LOG

Default Value: LINear

**[:SENSe]:BANDwidth[:RESolution]:AUTO <0 | 1 | ON | OFF>**
**[:SENSe]:BANDwidth[:RESolution]:AUTO?**

Title: RBW Auto

Description: Sets the state of the coupling of the resolution bandwidth to the frequency span. Setting the value to ON or 1 will result in the resolution bandwidth being coupled to the span. That is, when the span changes, the resolution bandwidth changes. Setting the value to OFF or 0 will result in the resolution bandwidth being decoupled from the span. That is, changing the span will not change the resolution bandwidth. When this command is issued, the resolution bandwidth setting itself will not change.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:BANDwidth:VIDeo:AUTO <0 | 1 | ON | OFF>**
**[:SENSe]:BANDwidth:VIDeo:AUTO?**

Title: VBW Auto

Description: Sets the state of the coupling of the video bandwidth to the resolution bandwidth. Setting the value to ON or 1 will result in the video bandwidth being coupled to the resolution bandwidth. That is, when the resolution bandwidth changes, the video bandwidth changes. Setting the value to OFF or 0 will result in the video bandwidth being decoupled from the resolution bandwidth. That is, changing the resolution bandwidth will not change the video bandwidth.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:BEAM <numeric_value>**
**[:SENSe]:BEAM?**

Title: Measured Beam

Description: Sets or queries the beam number to measure and display.

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 0

Range: 0 to 63

**[:SENSe]:BEAM:TRACking:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:BEAM:TRACking:STATe?**

Title: Detected Beam Tracking State

Description: Sets or queries the state of the beam tracking functionality.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:BPOWer:END <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**[:SENSe]:BPOWer:END?**

Title: End Time Burst Average Power

Description: Set and queries the end time position marker in burst power measurement.

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 40000 ms

Default Unit: ms

Range: 1e-05 ms to 60000 ms

**[:SENSe]:BPOWer:STARt <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**[:SENSe]:BPOWer:STARt?**

Title: Start Time Burst Average Power

Description: Set and queries the start time position marker in burst average power measurement.

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 10000 ms

Default Unit: ms

Range: 0 ms to 59999.999 ms

**[:SENSe]:BPOWer:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:BPOWer:STATe?**

Title: Burst Average Power State

Description: Sets and queries the state of the burst average power measurement

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:CAGGregation?**

Title: Get Carrier Aggregation Measurements

Description: Returns the Carrier Aggregation measurements for each carrier in a JSON format. Unlike [:SENSe]:FETCh:CAGGregation this command does not wait for the current measurement to complete before returning results.

**[:SENSe]:CAGGregation:CARRier:ACTive <numeric_value>**
**[:SENSe]:CAGGregation:CARRier:ACTive?**

Title: Active Carrier

Description: Sets and queries the active carrier in the Carrier Aggregation measurement.

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 1

Range: 1 to 8

**[:SENSe]:CAGGregation:CARRier:COUNt <numeric_value>**
**[:SENSe]:CAGGregation:CARRier:COUNt?**

Title: Number Of Carriers

Description: Sets and queries the number of carriers in the Carrier Aggregation measurement.

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 2

Range: 1 to 8

**[:SENSe]:CAGGregation:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:CAGGregation:STATe?**

Title: Carrier Aggregation Measurement State

Description: Turns on Carrier Aggregation, disables other measurements. When Carrier Aggregation measurement is on, changing frequencies, band, and antenna configurations will only apply to the active carrier. For instance, if the active carrier is set to 2, changing the center frequency will only change the center frequency for the carrier 2. Use command [:SENSe]:CAGGregation:CARRier:ACTive to set the active carrier.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:CBWidth <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe]:CBWidth?**

Title: Channel Bandwidth

Description: Sets or queries the channel bandwidth for the current demod measurement.

Full set of possible values are 5, 10, 15, 20, 25, 30, 40, 50, 60, 70, 80, 90, 100 MHz, though actual accepted values vary based on the current operating band and purchased options.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

Query Return: Numeric (Hz)

Default Value: 100000000 Hz

Default Unit: Hz

Range: 5000000 Hz to 400000000 Hz

**[:SENSe]:CBWidth <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe]:CBWidth?**

Title: Channel Bandwidth

Description: Sets or queries the channel bandwidth for the current demod measurement.

Full set of possible values are 1.4, 3, 5, 10, 15, 20MHz.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

Query Return: Numeric (Hz)

Default Value: 100000000 Hz

Default Unit: Hz

Range: 5000000 Hz to 400000000 Hz

**[:SENSe]:CBWidth:CATalog?**

Title: Channel Bandwidth Catalog

Description: Queries a list of all valid channel bandwidth values based on current settings.

Data is returned in a comma-separated list of frequencies in Hz, e.g. "5000000,10000000"

**[:SENSe]:CCHannel:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:CCHannel:STATe?**

Title: Control Channel Measurement State Command

Description: Turns on control channel measurement, disables other measurements

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:CHANnel:SCSPacing:CATalog?**

Title: Channel Subcarrier Spacing Catalog

Description: Queries a list of all valid channel subcarrier spacing values based on current settings.

Data is returned in a comma-separated list of frequencies in Hz, e.g. "15000,30000,120000"

Default Value: 15000,30000,120000

**[:SENSe]:CHPower:BANDwidth|BWIDth:INTegration <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe]:CHPower:BANDwidth|BWIDth:INTegration?**

Title: CHPWR Integration Bandwidth

Description: Sets the integration bandwidth for channel power measurement. Integration bandwidth must be less than or equal to span

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

Query Return: Numeric (Hz)

Default Value: 10350000 Hz

Default Unit: Hz

Range: 10 Hz to 400000000000 Hz

**[:SENSe]:CHPower:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:CHPower:STATe?**

Title: Channel Power State

Description: Sets the state of the channel power measurement, ON or OFF. When using :CONFigure:CHPower,the state is automatically set to ON

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:CINTerference:BANDwidth|BWIDth:INTegration <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe]:CINTerference:BANDwidth|BWIDth:INTegration?**

Title: CI Integration Bandwidth

Description: Sets and queries the integration bandwidth for Carrier to Interference Ratio measurement. Integration bandwidth must be less than or equal to span.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

Query Return: Numeric (Hz)

Default Value: 10350000 Hz

Default Unit: Hz

Range: 10 Hz to 400000000000 Hz

**[:SENSe]:CINTerference:CARRier:STATe?**

Title: CI Carrier State

Description: Queries the current measuring state for Carrier to Interference Ratio measurement. Values can be INACtive|ONCarrier|OFFCarrier, and are advanced with the MEAS:CINT:CONT command.

Query Return: INAC|ONC|OFFC

Default Value: INACtive

**[:SENSe]:CINTerference:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:CINTerference:STATe?**

Title: Carrier-to-Interference Ratio Measurement State

Description: Sets and queries the state of the Carrier-to-Interference Ratio measurement, ON or OFF. When using :CONFigure:CINTerference, the state is automatically set to ON.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:CLOCk:ADJust <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**[:SENSe]:CLOCk:ADJust?**

Title: Time Offset Correction

Description: This command is used to change the time offset correction for the GMC to GNSS calculation.

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 0 ms

Default Unit: ms

Range: -5 ms to 5 ms

**[:SENSe]:CONStellation <PBCH|PDSCh>**
**[:SENSe]:CONStellation?**

Title: Constellation Select

Description: Sets or queries the constellation data selection for the current LTE or 5G measurement. Only PBCH is available in 5G measurement.

Valid selections are: PBCH and PDSCH.

Parameter(s): <PBCH|PDSCh>

Query Return: PBCH|PDSC

Default Value: PDSCh

**[:SENSe]:CONStellation:PBCH?**

Title: PBCH Constellation

Description: Get the Physical Broadcast Channel (PBCH) constellation results immediately from the last complete measurement. See the description for :FETCh:CONStellation:PBCH for the results format.

**[:SENSe]:CONStellation:PBCH:BEAM <numeric_value>**
**[:SENSe]:CONStellation:PBCH:BEAM?**

Title: Constellation Measurement State

Description: Sets the beam number to filter for in the PBCH constellation measurement.

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 0

Range: 0 to 63

**[:SENSe]:CONStellation:PDSCh?**

Title: Data Channel Constellation

Description: Get the Physical Downlink Shared Channel (PDSCH) constellation results immediately from the last complete measurement. See the description for :FETCh:CONStellation:PDSCh for the results format.

**[:SENSe]:CONStellation:REFerence:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:CONStellation:REFerence:STATe?**

Title: Constellation Reference Point State

Description: Turns on/off constellation reference points

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:CONStellation:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:CONStellation:STATe?**

Title: Constellation Measurement State
Description: Turns on/off constellation measurement, disables other measurements
Parameter(s): <0 | 1 | ON | OFF>
Query Return: 0 | 1
Default Value: OFF

**[:SENSe]:CYCLicprefix <NORMal|EXTended|AUTO>**
**[:SENSe]:CYCLicprefix?**

Title: Cyclic Prefix
Description: Sets or queries the cyclic prefix type.
Available parameters are NORMal, EXTended, and AUTO for LTE mode.
Parameter(s): <NORMal | EXTended | AUTO>
Query Return: NORM | EXT | AUTO
Default Value: AUTO

**[:SENSe]:CYCLicprefix <NORMal|EXTended|AUTO>**
**[:SENSe]:CYCLicprefix?**

Title: Cyclic Prefix
Description: Sets or queries the cyclic prefix type. Only NORMal is available in 5GNR
Parameter(s): <NORMal | EXTended | AUTO>
Query Return: NORM | EXT | AUTO
Default Value: AUTO

**[:SENSe]:DEMod:DISCard:CSV**

Title: Discard Demod CSV
Description: This command discards the last file written for Demod CSV Recording.

**[:SENSe]:DEMod:GATE:DURation <COUPled|FRAMe>**
**[:SENSe]:DEMod:GATE:DURation?**

Title: Gated Spectrum Duration
Description: Set or query the gated duration in LTE or 5G frame power measurement. Valid selections are: COUPled - only symbols within the analysis view ([:SENSe]:FPAView) will be considered for gated spectrum. FRAMe - all symbols will be considered for gated spectrum.
Parameter(s): <COUPled | FRAMe>
Query Return: COUP | FRAM
Default Value: COUPled

**[:SENSe]:DEMod:GATE:SPECtrum:TYPe <UPLink|DWLink|FLEXible|GP|ALL|NONe>**
**[:SENSe]:DEMod:GATE:SPECtrum:TYPe?**

Title: Gated Spectrum Type

Description: Set or query the gated spectrum type in LTE or 5G frame power measurement. Valid selections are: UPLink, DWLink, GP(Guard Period), ALL and NONe in LTE. UPLink, DWLink, FLEXible, ALL and NONe in 5G.

Parameter(s): <UPLink|DWLink|FLEXible|GP|ALL|NONe>

Query Return: UPL|DWL|FLEX|GP|ALL|NON

Default Value: UPLink

**[:SENSe]:DEMod:LOG <LOG|PAUSe|STOP>**
**[:SENSe]:DEMod:LOG?**

Title: Demod Logging State

Description: Set or query what state Demod logging is currently in. LOG to collect data, PAUSe to stop taking data but keep the recording open, STOP to end the recording and save the csv file. Note that the resulting file will have some fields left unfilled depending on which measurement is being recorded. Such fields are recorded as nan for floating point numbers and as -1 for integers.

Parameter(s): <LOG|PAUSe|STOP>

Query Return: LOG|PAUS|STOP

Default Value: STOP

**[:SENSe]:DEMod:STORe:MSUS <string>**
**[:SENSe]:DEMod:STORe:MSUS?**

Title: Demod Log Storage Device

Description: This command sets or queries the storage device where the Demod CSV log files will be saved. Use the :MMEMory:CATalog:MSUSs? query to retrieve a list of valid parameters for this setting.

Parameter(s): <string>

Default Value: Internal

**[:SENSe]:DEMod:STORe:PATH <string>**
**[:SENSe]:DEMod:STORe:PATH?**

Title: Demod Log Location

Description: This command sets or queries the relative path to the directory where the Demod CSV log files will be saved. The log location is only used when Demod logging is enabled. The path is relative to the selected Demod storage device specified by the Device setting (:MMEMory:DEMod:STORe:MSUS).

Parameter(s): <string>

Default Value: Demod

**[:SENSe]:DENSity:ACQuisition <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**[:SENSe]:DENSity:ACQuisition?**

Title: Density Acquisition Time
Description: Density acquisition time for RTSA Mode
Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}
Query Return: Numeric (ms)
Default Value: 50 ms
Default Unit: ms
Range: 50 ms to 5000 ms

**[:SENSe]:DENSity:CSCale:AUTO <0 | 1 | ON | OFF>**
**[:SENSe]:DENSity:CSCale:AUTO?**

Title: RTSA Density Color Scale Auto
Description: In RTSA mode, this toggles the functionality for automatically scaling the color scale for the density spectrum measurement.
Parameter(s): <0 | 1 | ON | OFF>
Query Return: 0 | 1
Default Value: ON

**[:SENSe]:DENSity:CSCale:HIGH <numeric_value>**
**[:SENSe]:DENSity:CSCale:HIGH?**

Title: RTSA Density Color Scale High
Description: In RTSA mode, this indicates the percentage of hits needed at each point to be represented as the highest color of the density measurement's color scale (red).
Parameter(s): <numeric_value>
Query Return: Numeric
Default Value: 100
Range: 0 to 100

**[:SENSe]:DENSity:CSCale:LOW <numeric_value>**
**[:SENSe]:DENSity:CSCale:LOW?**

Title: RTSA Density Color Scale Low
Description: In RTSA mode, this indicates the percentage of hits needed at each point to be represented as the lowest color of the density measurement's color scale (blue).
Parameter(s): <numeric_value>
Query Return: Numeric
Default Value: 0
Range: 0 to 100

**[:SENSe]:DENSity:DETection:MINimum?**

Title: Minimum Detectable Signal Duration

Description: Gets the minimum duration (in ms) of a signal detectable by the system in RTSA mode.

Query Return: Numeric (ms)

Default Value: 5e-06 ms

Default Unit: ms

Range: 1e-06 ms to 5000000 ms

**[:SENSe]:DENSity:DISPlay:RESolution <NORMal|HIGH>**
**[:SENSe]:DENSity:DISPlay:RESolution?**

Title: Density Display Resolution

Description: This command sets the display resolution setting for the density measurement in RTSA mode.

Parameter(s): <NORMal|HIGH>

Query Return: NORM|HIGH

Default Value: HIGH

**[:SENSe]:DENSity:FFT:COUNt?**

Title: FFT Count

Description: Gets the current FFT count.

Query Return: Numeric

Default Value: 25000

Range: 32 to 4294967295

**[:SENSe]:DENSity:PERSistence <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**[:SENSe]:DENSity:PERSistence?**

Title: Density Persistence Time

Description: Sets the density persistence time parameter in RTSA mode.

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 0 ms

Default Unit: ms

Range: 0 ms to 10000 ms

**[:SENSe]:DENSity:PERSistence:CLEar**

Title: Density Persistence Clear

Description: Clears the persistent data for the denisty measurement in RTSA mode.

**[:SENSe]:DENSity:PERSistence:STATe <VARiable|INFinite>**
**[:SENSe]:DENSity:PERSistence:STATe?**

Title: Density Persistence State

Description: This command toggle the persistence setting for the density measurement in RTSA mode.

Parameter(s): <VARiable|INFinite>

Query Return: VAR|INF

Default Value: VARiable

**[:SENSe]:DENSity:POI?**

Title: Probability of Intercept

Description: Gets the current POI value.

Query Return: Numeric (ms)

Default Value: 0.004 ms

Default Unit: ms

Range: 1e-06 ms to 5000000 ms

**[:SENSe]:DENSity:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:DENSity:STATe?**

Title: Density Spectrum State

Description: Sets the state of the density spectrum measurement display.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:DETector[:FUNCtion] <POSitive|RMS|NEGative|SAMPle|NORMal>**
**[:SENSe]:DETector[:FUNCtion]?**

Title: Detection Mode

Description: Sets the detection method for calculating each display point. This command changes the detection method for the currently selected trace (see Select Trace command). To set the detection mode for any trace use the Trace Detector Type command. Each display point represents several measurements. The detection type determines how the display point is derived from its associated measurements. POSitive Peak detection displays the maximum value of the associated measurements. RMS detection displays the average power of the associated measurement. NEGative Peak detection displays the minimum value of the associated measurements. SAMPle detection selects one of the associated measurement points to display. In zero span and RTSA mode the SAMPle detection replaces RMS detection. NORMal detection displays the maximum value, POSitive Peak, when the point is odd and minimum value, NEGative Peak, when the point is even.

Parameter(s): <POSitive|RMS|NEGative|SAMPle|NORMal>

**[:SENSe]:DF:INFormation?**

Title: Directional Antenna Information

Description: Requests combined data from the GPS module, the compass, and the status of the trigger switch, RF amp and the overload detector.

Response:
llll.lll,ns,nnnn.nnn,ew,q,s,aa.a,st,ss.ss,ttt.tt,ddd.d,dew,bbb.b,pp.p,rr.r,ol,trg,rfa

llll.lll: Latitude in degrees and decimal minutes, e.g. 3708.806 is 37 degrees and 8.806 minutes or 8 minutes and 48.36 seconds.

ns: Latitude N (North) or S (South).

nnnn.nnn: Longitude in degrees and decimal minutes.

ew: East or West

q: GPS quality: 0 = no GPS, 1 = GPS, 2 = DGPS/SBAS.

s: satellites

aa.a: Altitude in meters.

st = RMC Status: A = valid, V = navigation receiver warning.

ss.ss: Speed over ground in knots.

tt.tt: Track made good in degrees true (true bearing).

ddd.d : Magnetic declination in degrees.

dew: Magnetic declination E (east) or W (West).

bbb.b: Compass bearing in degrees 0.0 to 359.9.

pp.p: Pitch in degrees 0.0 to +/- 90.0.

rr.r: Roll in degrees 0.0 to +/- 90.0.

ol: Overload detector status, 0 = no overload, 1 = overload.

trg: Trigger switch status, 0 = switch open, 1 = switch closed.

rfa: RF preamp status, 0 = RF preamp off, 1 = RF preamp on.

**[:SENSe]:DUPLex <FDD|TDD>**
**[:SENSe]:DUPLex?**

Title: Duplex Type

Description: Sets or queries the duplex type. TDD is available for 5GNR and LTE. FDD is availiable for LTE.

Parameter(s): <FDD|TDD>

Query Return: FDD|TDD

Default Value: TDD

**[:SENSe]:EARFchannel <numeric_value>**
**[:SENSe]:EARFchannel?**

Title: E-UTRA Absolute RF Channel Number

Description: Sets or queries the E-UTRA Absolute RF Channel Number for the current demod measurement. Accepted values vary based on the current operating band. Inputting invalid values will set the closest valid value possible for the given circumstances.

When the operating band is not being used ([:SENSe]:OBANd is INValid), this value will not be settable. Attempts to set this when the system is in this state will return the error -221.

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 0

Range: 0 to 3279165

**[:SENSe]:EIRPower:CLOSs <numeric_value> {DB}**
**[:SENSe]:EIRPower:CLOSs?**

Title: EIRP Test Port Cable Loss

Description: This command sets or queries the test port cable loss for EIRP measurement.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {DB}

Query Return: Numeric (dB)

Default Value: 0 dB

Default Unit: dB

Range: 0 dB to 999 dB

**[:SENSe]:EIRPower:DISTance <numeric_value>**
**[:SENSe]:EIRPower:DISTance?**

Title: EIRP Measurement's Distance

Description: This command sets or queries the EIRP measurement's distance. Allowed unit types include m (meters) or ft (feet).

Note that if no unit is passed in, the instrument will assume that the incoming or outgoing unit is the unit type set through the [:SENSe]:EIRPower:DISTance:UNIT SCPI command.

Parameter(s): <numeric_value>

Default Value: 0.01 m

Default Unit: m

Range: 0.01 m to 50000 m

**[:SENSe]:EIRPower:DISTance:UNIT <METers|FEET>**
**[:SENSe]:EIRPower:DISTance:UNIT?**

Title: EIRP Distance Units

Description: This command sets or queries the EIRP distance units. The query returns:

MET - Meters

FEET - Feet

Parameter(s): <METers | FEET>

Query Return: MET | FEET

Default Value: METers

**[:SENSe]:EIRPower:MAXimum:COUNt?**

Title: Maximum EIRP Count

Description: This command returns the current maximum EIRP count.

**[:SENSe]:EIRPower:MAXimum:RESet**

Title: Reset Maximum EIRP Count

Description: This command resets the maximum EIRP count.

**[:SENSe]:EIRPower:PLOSs?**

Title: EIRP Path Loss

Description: This command queries the EIRP total path loss between the base station gNB and test antenna.

Query Return: Numeric (dB)

Default Value: 0 dB

Default Unit: dB

Range: 0 dB to 999 dB

**[:SENSe]:EIRPower:RAGain <numeric_value> {DB}**
**[:SENSe]:EIRPower:RAGain?**

Title: EIRP Rx Antenna Gain

Description: This command sets or queries the EIRP Rx Antenna Gain in dBi.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {DB}

Query Return: Numeric (dB)

Default Value: 0 dB

Default Unit: dB

Range: 0 dB to 999 dB

**[:SENSe]:EIRPower:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:EIRPower:STATe?**

Title: EIRP State

Description: Sets the state of the EIRP measurement, ON or OFF. When using :CONFigure:EIRPower,the state is automatically set to ON

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:EIRPower:SUM:RESet**

Title: EIRP Sum Reset

Description: Resets the Horizontal and Vertical EIRP values.

**[:SENSe]:EIRPower:TAGain <numeric_value> {DB}**
**[:SENSe]:EIRPower:TAGain?**

Title: EIRP Tx Antenna Gain

Description: This command sets or queries the EIRP Tx Antenna Gain in dBi.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {DB}

Query Return: Numeric (dB)

Default Value: 0 dB

Default Unit: dB

Range: 0 dB to 999 dB

**[:SENSe]:FETch:THD?**

Title: Get THD Measurement

Description: Returns the THD measurements result. This command returns the fundamental frequency, harmonics and total harmonics distortion info in json format. If a measurement is in progress, it blocks until complete. An example is as follows:

{

"Fundamental":{"Frequency":751137724.5508982, "Index":6,"Level":-71.11, "Level dBc":null},

"Harmonics":[{"Frequency":752275449.1017964,"Index":12, "Level":-60.63000000000001, "Level dBc":-0.0500},

{"Frequency":753413173.6526946,"Index":18,"Level":-64.86, "Level dBc":-0.0670},

{"Frequency":754550898.2035928,"Index":24,"Level":-74.82, "Level dBc":-0.0710},

{"Frequency":755688622.754491,"Index":30,"Level":-63.31 ,"Level dBc":-0.0520},

{"Frequency":756826347.3053892,"Index":36,"Level":-64.23, "Level dBc":-0.0550},

{"Frequency":757964071.8562875,"Index":42,"Level":-65.29, "Level dBc":-0.0600}],

"THD":[{"Name":"THD","Units":"dB","Value":-0.1842652869500627},

{"Name":"THD","Units":"%","Value":97.90091158122401}]

}

**[:SENSe]:FPAView <FRAMe|SUBFrame|SLOT>**
**[:SENSe]:FPAView?**

Title: Frame Power Analysis View

Description: Sets or queries the Frame Power Analysis view mode for the current LTE measurement.
Valid selections are: FRAME, SUBFRAME or SLOT.

Parameter(s): <FRAMe|SUBFrame|SLOT>

Query Return: FRAM|SUBF|SLOT

Default Value: FRAMe

**[:SENSe]:FRAMe:POWer:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:FRAMe:POWer:STATe?**

Title: Frame Power Analysis State Command

Description: Turns on frame power analysis measurement, disables other measurements

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:FREQuency:CENTer <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe]:FREQuency:CENTer?**

Title: Center Frequency

Description: Sets the center frequency. Note that changing the value of the center frequency will change the value of the coupled parameters Start Frequency and Stop Frequency. It may also change the value of the span.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

Query Return: Numeric (Hz)

Default Value: 3000000000 Hz

Default Unit: Hz

Range: -99999999995 Hz to 299999999995 Hz

**[:SENSe]:FREQuency:CENTer:SHARe <0 | 1 | ON | OFF>**
**[:SENSe]:FREQuency:CENTer:SHARe?**

Title: Share Center Frequency

Description: Enables "sharing" of the instrument center frequency.
When ON, changing any center frequency setting attempts
to change every other instance.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:FREQuency:LIMit?**

Title: Device Frequency Limit

Description: Query the devices maximum stop frequency.

Query Return: Numeric (Hz)

Default Value: 300000000000 Hz

Default Unit: Hz

Range: 0 Hz to 300000000000 Hz

**[:SENSe]:FREQuency:OFFSet <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe]:FREQuency:OFFSet?**

Title: Frequency Offset

Description: Set a frequency offset, which will be added to the start, stop, and center frequencies. This offset is for display purposes only and does not affect the frequency range being measured.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

Query Return: Numeric (Hz)

Default Value: 0 Hz

Default Unit: Hz

Range: -10000000000 Hz to 10000000000 Hz

**[:SENSe]:FREQuency:SPAN <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe]:FREQuency:SPAN?**

Title: Span

Description: Sets the frequency span. Setting the value of <freq> to 0 Hz is the equivalent of setting the span mode to zero span. Note that changing the value of the frequency span will change the value of the coupled parameters Start Frequency and Stop Frequency and may change the Center Frequency.

Note that this command is not available for Spectral Emission Mask measurement.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

Query Return: Numeric (Hz)

Default Value: 400000000000 Hz

Default Unit: Hz

Range: 10 Hz to 400000000000 Hz

**[:SENSe]:FREQuency:SPAN:FULL**

Title: Set to Full Span

Description: Sets the frequency span to full span. Note that changing the value of the frequency span will change the value of the coupled parameters, Start Frequency and Stop Frequency and may change the Center Frequency.

**[:SENSe]:FREQuency:SPAN:LAST**

Title: Set to Last Span

Description: Sets the frequency span to the previous span value. Note that changing the value of the frequency span will change the value of the coupled parameters, Start Frequency and Stop Frequency and may change the Center Frequency.

**[:SENSe]:FREQuency:STARt <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe]:FREQuency:STARt?**

Title: Start Frequency

Description: Sets the start frequency. Note that in the spectrum analyzer, changing the value of the start frequency will change the value of the coupled parameters, Center Frequency and Span.

Note that this command is not available for Spectral Emission Mask measurement.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

Query Return: Numeric (Hz)

Default Value: 0 Hz

Default Unit: Hz

Range: -100000000000 Hz to 299999999990 Hz

**[:SENSe]:FREQuency:STEP[:INCRement] <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe]:FREQuency:STEP[:INCRement]?**

Title: Frequency Step

Description: Set or query the step size to gradually increase or decrease frequency value. This command doesn't change any behavior in the backend but will be included in save/recall operations.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

Query Return: Numeric (Hz)

Default Value: 1000000 Hz

Default Unit: Hz

Range: 0.1 Hz to 1000000000 Hz

**[:SENSe]:FREQuency:STOP <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe]:FREQuency:STOP?**

Title: Stop Frequency

Description: Sets the stop frequency. Note that in the spectrum analyzer, changing the value of the stop frequency will change the value of the coupled parameters, Center Frequency and Span.

Note that this command is not available for Spectral Emission Mask measurement.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

Query Return: Numeric (Hz)

Default Value: 6000000000 Hz

Default Unit: Hz

Range: -99999999990 Hz to 300000000000 Hz

**[:SENSe]:FREQuency:SWEep:TIME <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**[:SENSe]:FREQuency:SWEep:TIME?**

Title: Frequency Sweep Time

Description: Sets the value of the minimum frequency sweep time parameter. The sweep will complete in the shortest time possible greater than the specified time. This value is ignored when Frequency Sweep Time Auto is set to TRUE. See command :FREQuency:SWEep:TIMe:AUTO. Any changes to this setting would automatically set Frequency Sweep Time Auto to false.

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 5000 ms

Default Unit: ms

Range: 0.001 ms to 3600000 ms

**[:SENSe]:FREQuency:SWEep:TIMe:AUTO <0 | 1 | ON | OFF>**
**[:SENSe]:FREQuency:SWEep:TIMe:AUTO?**

Title: Frequency Sweep Time Auto

Description: Specifies Sweep Time State as it applies to the Frequency Sweep Time. The default state is set to be TRUE and does not use the Frequency Sweep Time to determine the sweep speed.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:FSTRength:ANTenna <string>**
**[:SENSe]:FSTRength:ANTenna?**

Title: Field Strength Antenna

Description: Select or query the current antenna set for Field Strength.

Parameter(s): <string>

Default Value: None

**[:SENSe]:FSTRength:ANTenna:FACTor?**

Title: Field Strength Antenna Factor

Description: Returns an interpolated list of antenna factors as a json array where the size of the array corresponds to the number of display points currently selected.

For example: [ 2.0, 3.4, 5.3 ]

**[:SENSe]:FSTRength:ANTenna:FILE <block data>**
**[:SENSe]:FSTRength:ANTenna:FILE?**

Title: User Antenna List File Import and Export

Description: This command imports/exports a file to/from the instrument. Data is transferred to/from the instrument as an IEEE definite length arbitrary block response, which has the form <header><block>.

Block Data: The data to be written to the instrument in block data format

Please note that the maximum file transfer size to the instrument is 25 MB.

The ASCII header specifies the number of data bytes of the file. It looks like #AX, where A is the number of digits in X and X is the number of bytes in the <block>. The first character is the literal ascii hash '#' 043. The second character (A) is a single ascii digit '1' to '9' describing the number of bytes in the length section (X). This number is called nlength. The next nlength bytes make up an ascii string of digits '1' to '9' describing the length of the <block> data.

For example, if the first 6 bytes are #49999, then the nlength is 4. The 4 bytes of length are 9999. After that follows the <block>, which would be 9999 bytes in size.

The file is returned in block data format with an ASCII header The query command will return a #10 if the file cannot be found.

Parameter(s): <block data>

**[:SENSe]:FSTRength:ANTenna:LIST?**

Title: Field Strength Antenna List

Description: Returns the list of available antennas. Return value is json array containing the Antenna name, its Start Frequency and its Stop Frequency.

For example:

[{

"name": "Anritsu_Antenna_1",

"startFreq":1710000000,

"stopFreq":1880000000

},

{

"name": "Anritsu_Antenna_2",

"startFreq":1850000000,

"stopFreq":1990000000

}]

**[:SENSe]:FSTRength:ANTenna:LIST:USER?**

Title: Field Strength User Antenna List

Description: Returns the list of available user antennas. Return value is json array containing the Antenna name, its Start Frequency and its Stop Frequency.

For example:

[{

"name": "User_Antenna_1",

"startFreq":1950000000,

"stopFreq":2500000000

},

{

"name": "User_Antenna_2",

"startFreq":1650000000,

"stopFreq":1850000000

}]

**[:SENSe]:FSTRength:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:FSTRength:STATe?**

Title: Field Strength State

Description: Sets the state of the field strength measurement ON or OFF.

If this command is set to ON, the amplitude measurement units on the instrument will be converted to its Field Strength value.

To query the amplitude measurement units, use the :UNIT:POWer? query command.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:FSTRength:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:FSTRength:STATe?**

Title: Field Strength State

Description: Sets the state of the field strength measurement ON or OFF.

If this command is set to ON, the amplitude measurement units on the instrument will be converted to its Field Strength value.

To query the amplitude measurement units, use the :UNIT:POWer? query command.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:GATE:DELay <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**[:SENSe]:GATE:DELay?**

Title: Gate Delay

Description: Sets or queries the gate delay time. Available values are limited based on current mode.

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 0 ms

Default Unit: ms

Range: 0 ms to 199.999 ms

**[:SENSe]:GATE:DLSLots[:ONE] <numeric_value>**
**[:SENSe]:GATE:DLSLots[:ONE]?**

Title: P1 Downlink Slots

Description: Sets or queries the number of Downlink slots for P1 (the first section of the pattern). Per 3GPP TS 38.213 (11.1 Slot configuration), within the standard defined structure IE tdd-UL-DL-ConfigurationCommon, "nrofDownlinkSlots" designates the number of downlink slots - slots composed of only downlink symbols - at the start of a given period, in the range (0..MaxNrofSlots).

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 3

Range: 0 to 160

**[:SENSe]:GATE:DLSLots:TWO <numeric_value>**
**[:SENSe]:GATE:DLSLots:TWO?**

Title: P2 Downlink Slots

Description: Sets or queries the number of Downlink slots for P2 (the optional second section of the pattern). Per 3GPP TS 38.213 (11.1 Slot configuration), within the standard defined structure IE tdd-UL-DL-ConfigurationCommon, "nrofDownlinkSlots" designates the number of downlink slots - slots composed of only downlink symbols - at the start of a given period, in the range (0..MaxNrofSlots).

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 4

Range: 0 to 160

**[:SENSe]:GATE:DLSYmbols[:ONE] <numeric_value>**
**[:SENSe]:GATE:DLSYmbols[:ONE]?**

Title: P1 Downlink Symbols

Description: Sets or queries the number of Downlink symbols for P1 (the first section of the pattern). Per 3GPP TS 38.213 (11.1 Slot configuration), within the standard defined structure IE tdd-UL-DL-ConfigurationCommon, "nrofDownlinkSymbols" defines the number of downlink symbols within the slot following the downlink slots of a given period, in the range (0..MaxNrofSymbols-1).

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 6

Range: 0 to 13

**[:SENSe]:GATE:DLSYmbols:TWO <numeric_value>**
**[:SENSe]:GATE:DLSYmbols:TWO?**

Title: P2 Downlink Symbols

Description: Sets or queries the number of Downlink symbols for P2 (the optional second section of the pattern). Per 3GPP TS 38.213 (11.1 Slot configuration), within the standard defined structure IE tdd-UL-DL-ConfigurationCommon, "nrofDownlinkSymbols" defines the number of downlink symbols within the slot following the downlink slots of a given period, in the range (0..MaxNrofSymbols-1).

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 0

Range: 0 to 13

**[:SENSe]:GATE:FRAMe:OFFSet <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**[:SENSe]:GATE:FRAMe:OFFSet?**

Title: Frame Time Offset

Description: Specify a Custom Frame Time Offset to be used when the Frame Start Time is set to Custom

ex: sense:gate:frame:starttime custom

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 0 ms

Default Unit: ms

Range: -4.999999 ms to 5 ms

**[:SENSe]:GATE:FRAMe:STARttime <AUTO|SYNO|UTC|UTC3pos|UTC2neg|CUSTom>**
**[:SENSe]:GATE:FRAMe:STARttime?**

Title: Frame Start Time

Description: UTC = Coordinated Universal Time in accordance with Recommendation ITU-R TF.460. The typical measuring signal is 1 pps (pulse-per-second) signal from the clock device with the rising-edge materializing the exact second epoch. The start of frame defined as the beginning of the first slot in the UL/DL pattern has to be phase-synchronised with the frame start time. The start of the radio frame on the output shall be synchronous with the input time reference, i.e., when an UTC traceable reference is required, the start of the radio frame shall be aligned with the start time of the UTC second epoch. A configurable time-offset of start frame shall be supported by all base stations in synchronized TDD-unicast areas in order to achieve interoperability in coexistence scenarios.

AUTO, indicates use of automatic synchronization for all TDD frame start times.

SYNO, or Synchronize Once, indicates use of automatic synchronization once to obtain a TDD frame start time, to be used for successive frames. Setup parameter changes will automatically force a re-synchronization. You can also send [:SENSe]:GATE:FRAMe:STARttime:FORCe to force a re-synchronization.

UTC, represents a TDD frame start time which corresponds with the Coordinated Universal Time.

UTC3pos, represents a frame start time of UTC +3 ms. (5G only)

UTC2neg, represents a frame start time of UTC -2 ms. (5G only)

CUSTom, represents a custom frame start time in the range (-5, 5] ms. To specify this value, see the "Frame Time Offset" command. (sense:gate:frame:offset)

Parameter(s): <AUTO | SYNO | UTC | UTC3pos | UTC2neg | CUSTom>

Query Return: AUTO | SYNO | UTC | UTC3 | UTC2 | CUST

Default Value: SYNO

**[:SENSe]:GATE:FRAMe:STARttime:FORCe**

Title: Frame Force synchronization

Description: When SYNO is set with this command [:SENSe]:GATE:FRAMe:STARttime, synchronization will be attempted until it is successful. The frame timing obtained from the successful synchronization is used for subsequent UL/DL Interference measurements. Sending this command will force a re-synchronization so an updated frame timing can be used for the same. This could be used if the signal condition changes or we need to re-establish sync with another base station.

**[:SENSe]:GATE:FRAMestructure <A|B1|B2|CUSTom>**
**[:SENSe]:GATE:FRAMestructure?**

Title: TDD MFCN Frame Structure

Description: Sets or queries use of either a Custom Frame Structure or one of two recommended by the Electronic Communications Committee (ECC), for time division duplexing (TDD) of mobile/fixed communications network (MFCN) operation to facilitate cross-border coordination of TDD MFCN in the 3400-3800 MHz frequency band.

Parameter(s): <A | B1 | B2 | CUSTom>

Query Return: A | B1 | B2 | CUST

Default Value: B1

**[:SENSe]:GATE:FRAMestructure:SSTYpe <TYP1|TYP2>**
**[:SENSe]:GATE:FRAMestructure:SSTYpe?**

Title: TDD MFCN Frame B Special Slot Type

Description: Sets or queries special slot configuration type for Frame B1 and B2 structures of [:SENSe]:GATE:FRAMestructure selections. Valid selections are: TYP1 for D:G:U 6:4:4, or TYP2 for D:G:U 4:6:4. \

Parameter(s): <TYP1|TYP2>

Query Return: TYP1|TYP2

Default Value: TYP1

**[:SENSe]:GATE:LENGth <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**[:SENSe]:GATE:LENGth?**

Title: Gate Delay Length

Description: Sets or queries the length of the gate delay. Avalailable values are limited based on current mode.

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 1 ms

Default Unit: ms

Range: 0.001 ms to 200 ms

**[:SENSe]:GATE:PERiod <P1s|P20ms|P10ms>**
**[:SENSe]:GATE:PERiod?**

Title: Gated Sweep Period Length

Description: Sets or queries the time period used for the Gated Sweep functionality. Available settings are: "P10ms" (10ms) and "P20ms" (20ms).

Parameter(s): <P1s|P20ms|P10ms>

Query Return: P1|P20|P10

Default Value: P20ms

**[:SENSe]:GATE:PERIodicity[:ONE]**
**<M0P5|M0P625|M1P0|M1P25|M2P0|M2P5|M3P0|M4P0|M5P0|M10P0|NONE>**
**[:SENSe]:GATE:PERIodicity[:ONE]?**

Title: P1 Transmission Periodicity

Description: Sets or queries the Transmission Periodicity for P1 (the first section of the pattern). Per 3GPP TS 38.213 (11.1 Slot configuration), within the standard defined structure IE tdd-UL-DL-ConfigurationCommon, "dl-UL-TransmissionPeriodicity" defines the slot configuration period of a given length in msec.

The designations are split into a whole msec component preceeded by M and a fractional msec component preceeded by P. Alternatively, set NONE to set the length of P1 to zero.

Resulting Time Durations(ms): 0.5 0.625, 1, 1.25, 2, 2.5, 5, 10

Parameter(s): <M0P5|M0P625|M1P0|M1P25|M2P0|M2P5|M3P0|M4P0|M5P0|M10P0|NONE>

Query Return: M0P5|M0P625|M1P0|M1P25|M2P0|M2P5|M3P0|M4P0|M5P0|M10P0|NONE

Default Value: M3P0

**[:SENSe]:GATE:PERIodicity:TWO**
**<M0P5|M0P625|M1P0|M1P25|M2P0|M2P5|M3P0|M4P0|M5P0|M10P0|NONE>**
**[:SENSe]:GATE:PERIodicity:TWO?**

Title: P2 Transmission Periodicity

Description: Sets or queries the Transmission Periodicity for P2 (the optional second section of the pattern). Per 3GPP TS 38.213 (11.1 Slot configuration), within the standard defined structure IE tdd-UL-DL-ConfigurationCommon, "dl-UL-TransmissionPeriodicity" defines the slot configuration period of a given length in msec.

The designations are split into a whole msec component preceeded by M and a fractional msec component preceeded by P. Alternatively, set NONE to set the length of P2 to zero.

Resulting Time Durations(ms): 0.5 0.625, 1, 1.25, 2, 2.5, 5, 10

Parameter(s): <M0P5|M0P625|M1P0|M1P25|M2P0|M2P5|M3P0|M4P0|M5P0|M10P0|NONE>

Query Return: M0P5|M0P625|M1P0|M1P25|M2P0|M2P5|M3P0|M4P0|M5P0|M10P0|NONE

Default Value: M2P0

**[:SENSe]:GATE:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:GATE:STATe?**

Title: Gated Sweep State

Description: Sets or queries the status of the Gated Sweep functionality. Usage requires Option 0090.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:GATE:TRIGger <GPS|EXT1|EXT2|EXT>**
**[:SENSe]:GATE:TRIGger?**

Title: Gated Sweep Trigger Source

Description: Sets or queries the trigger source used for the Gated Sweep functionality.

Parameter(s): <GPS|EXT1|EXT2|EXT>

Query Return: GPS|EXT1|EXT2|EXT

Default Value: GPS

**[:SENSe]:GATE:ULSLots[:ONE] <numeric_value>**
**[:SENSe]:GATE:ULSLots[:ONE]?**

Title: P1 Uplink Slots

Description: Sets or queries the number of Uplink slots for P1 (the first section of the pattern). Per 3GPP TS 38.213 (11.1 Slot configuration), within the standard defined structure IE tdd-UL-DL-ConfigurationCommon, "nrofUplinkSlots" defines the number of uplink slots - slots composed of only uplink symbols - at the end of a given period, in the range (0..MaxNrofSlots).

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 2

Range: 0 to 160

**[:SENSe]:GATE:ULSLots:TWO <numeric_value>**
**[:SENSe]:GATE:ULSLots:TWO?**

Title: P2 Uplink Slots

Description: Sets or queries the number of Uplink slots for P2 (the optional second section of the pattern). Per 3GPP TS 38.213 (11.1 Slot configuration), within the standard defined structure IE tdd-UL-DL-ConfigurationCommon, "nrofUplinkSlots" defines the number of uplink slots - slots composed of only uplink symbols - at the end of a given period, in the range (0..MaxNrofSlots).

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 0

Range: 0 to 160

**[:SENSe]:GATE:ULSYmbols[:ONE] <numeric_value>**
**[:SENSe]:GATE:ULSYmbols[:ONE]?**

Title: P1 Uplink Symbols

Description: Sets or queries the number of Uplink symbols for P1 (the first section of the pattern). Per 3GPP TS 38.213 (11.1 Slot configuration), within the standard defined structure IE tdd-UL-DL-ConfigurationCommon, "nrofUplinkSymbols" defines the number of uplink symbols within the slot preceding the uplink slots in a given period, in the range (0..MaxNrofSymbols-1).

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 4

Range: 0 to 13

**[:SENSe]:GATE:ULSYmbols:TWO <numeric_value>**
**[:SENSe]:GATE:ULSYmbols:TWO?**

Title: P2 Uplink Symbols

Description: Sets or queries the number of Uplink symbols for P2 (the optional second section of the pattern). Per 3GPP TS 38.213 (11.1 Slot configuration), within the standard defined structure IE tdd-UL-DL-ConfigurationCommon, "nrofUplinkSymbols" defines the number of uplink symbols within the slot preceding the uplink slots in a given period, in the range (0..MaxNrofSymbols-1).

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 0

Range: 0 to 13

**[:SENSe]:GSCNumber <numeric_value>**
**[:SENSe]:GSCNumber?**

Title: Global Sync Channel Number

Description: Sets or queries the Global Sync Channel Number for the current demod measurement. Accepted values vary based on the current operating band and absolute RF channel number. Inputting invalid values will set the closest valid value possible for the given circumstances.

When the operating band is not being used ([:SENSe]:OBANd is INValid), this value will not be settable. Attempts to set this when the system is in this state will return the error -221.

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 2

Range: 2 to 26639

**[:SENSe]:IQ:BITS <F32|I32|I16|A16|I10|I8>**
**[:SENSe]:IQ:BITS?**

Title: IQ Sample Format

Description: The number of IQ bits per sample. Lower values enable higher throughput (continuous IQ capture) or longer maximum capture length (block IQ capture). The number of samples per frame increases as the bits per sample decreases:

Value Bits per sample Samples per frame Type

F32 32 1 (IQ) floating point

I32 32 1 (IQ) integer

I16 16 2 (IIQQ) integer

A16 16 2 (IQIQ) integer

I10 10 3 (III_QQQ_) integer

I8 8 4 (IIIIQQQQ) integer

Parameter(s): <F32|I32|I16|A16|I10|I8>

Query Return: F32|I32|I16|A16|I10|I8

Default Value: F32

**[:SENSe]:IQ:CONCatenate[:STATe] <0 | 1 | ON | OFF>**
**[:SENSe]:IQ:CONCatenate[:STATe]?**

Title: IQ Stream Concatenate State

Description: Determines whether IQ data files produced during streaming are combined into one file when streaming ends. If set to ON, IQ streams will result in one data file. If set to OFF, IQ streams will result in many individual files with indices in the file names to indicate order. The many individual files will be grouped into a folder named with the datetime of the start of the stream. Setting this to OFF can significantly improve file saving times.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:IQ:DISCard**

Title: IQ Discard

Description: Stops any IQ capture or stream in progress and discards all data.

**[:SENSe]:IQ:FPGA:SWITch <PCIe|IQC>**

Title: Swtich FPGA Image to Configure Data Out Port

Description: This command switchs the FPGA image to configure Data Out port for IQ data streaming.

Options:

PCIe: Streaming IQ data though Data Out port without data conversion.

IQC: Streaming IQ data to data converter device though Data Out port.

Parameter(s): <PCIe|IQC>

**[:SENSe]:IQ:LENGth <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**[:SENSe]:IQ:LENGth?**

Title: Capture Length

Description: The set form of this command sets the iq length in milliseconds and query form returns the iq length in milliseconds

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 10000 ms

Default Unit: ms

Range: 0.001 ms to 100000000 ms

**[:SENSe]:IQ:LENGth:MAX?**

Title: IQ Capture Max Length

Description: This command returns max IQ Capture length due to available memory space in the selected storage device.

Query Return: Numeric (ms)

Default Value: 0.001 ms

Default Unit: ms

Range: 0 ms to 90000000 ms

**[:SENSe]:IQ:MEMory?**

Title: IQ Available Memory

Description: This command returns available memory size for IQ Capture in the selected storage device. The value may be less than the free space on the selected storage device because it's limited to the size of the IQ data buffer.

Query Return: Numeric

Default Value: 0

Range: 0 to INFINITY

**[:SENSe]:IQ:MEMory:STREam?**

Title: IQ Stream Available Memory

Description: This command returns available memory size for IQ Stream in the selected storage device.

Query Return: Numeric

Default Value: 0

Range: 0 to INFINITY

### [:SENSe]:IQ:METadata?

Title: Query IQ Data

Description: Queries the current Metadata of ongoing or previosuly completed capture/stream.

If no capture/stream has been initiated since instrument has been powered on, a data out of range error will be added to the SCPI error queue. If the command is sent before some data has been populated, some fields will show as "nan" or "Unknown".

An example of returned IQ capture/stream data is as follows: <?xml version="1.0" encoding="UTF-8"?>

<metadata>

<DataFile Value="Capture_20200731T114504.278_225381550ns_37850040000_1e+8.dgz"/>

<Vendor Value="Anritsu"/>

<Model Value="MS2090A"/>

<CenterFrequency Value="37850040000" Units="Hz"/>

<CaptureBandwidth Value="1.1e+8" Units="Hz"/>

<SampleRate Value="2e+8" Units="Hz"/>

<Span Value="1e+8" Units="Hz"/>

<ReferenceLevel Value="-40" Units="dBm"/>

<Attenuation Value="0" Units="dB"/>

<PreampState Value="1"/>

<IFFilterBandwidth Value="750e+3" Units="Hz"/>

<TriggerType Value="SING"/>

<TriggerSource Value="IMM"/>

<TriggerTime Value="0"/>

<TriggerOffset Value="0"/>

<TriggerDelay Value="0" Units="µs"/>

<Latitude Value="32.7712"/>

<Longitude Value="-117.146"/>

<GpsFix Value="1"/>

<CaptureStart Value="2020-07-31T11:45:04"/>

<ClockOffset Value="2.25382e+8"/>

<ScaleFactor Value="0.316228"/>

<BytesPerFrame Value="8"/>

<BitsPerSample Value="32"/>

<EmbeddedTimestamp Value="1"/>

<LittleEndian Value="1"/>

<IQInterleave Value="1"/>

<SignedValues Value="1"/>

<DataEncoding Value="F32"/>

<IQReversed Value="0"/>

<SampleCount Value="8000000"/>

<HardwareErrors ADCOverRange="0" DeviceAccess="0" EEPROM="0"
IrregularData="0" Limit="0" LockLO1="0" LockLO2="0"

NoCal="0" Overpower="0" SLOLock="0" Saturation="0" TgLockLo1="0"/>

</metadata>

**[:SENSe]:IQ:MODE <SINGle|STREam|CONTinuous>**
**[:SENSe]:IQ:MODE?**

Title: Capture Mode

Description: Specifies the type of capture mode

1.Single mode does a single block capture

2.Streaming mode does real time streaming capture

3.Continuous mode takes consecutive single block captures

Parameter(s): <SINGle|STREam|CONTinuous>

Query Return: SING|STRE|CONT

Default Value: SINGle

**[:SENSe]:IQ:PORT <DATaout|USB|ETHernet>**
**[:SENSe]:IQ:PORT?**

Title: IQ Streaming Output Port Type

Description: This command sets or queries the IQ Streaming output port type.

DATaout will stream IQ data to the Data Out port.

USB will stream IQ data to the USB device.

ETHernet will stream IQ data throgh TCP/IP.

Parameter(s): <DATaout|USB|ETHernet>

Query Return: DAT|USB|ETH

Default Value: ETHernet

**[:SENSe]:IQ:PORT:DATAout?**

Title: IQ DataOut Type

Description: This command queries the Data Out port type.

IQC will stream IQ data to a data converter device using the Data Out port.

PCIe will stream IQ data though the Data Out port without conversion.

The data out port can be enabled using IQ:PORT command.

Query Return: IQC|PCI

Default Value: IQC

**[:SENSe]:IQ:SAMPle**
**<SB0|SB1|SB2|SB3|SB4|SB5|SB6|SB7|SB8|SB9|SB10|SB11|SB12|SB13|SB14|SB15|SB16|SB17|SB18|SB19|SB20>**
**[:SENSe]:IQ:SAMPle?**

Title: IQ Sample Rate and Bandwidth

Description: This command sets or queries the sample rate and bandwidth value pair for IQ Capture or IQ Streaming. For model MS2090A the IF filter has a nominal pass band of 120 MHz. SB0 is only available with option 0105.

For model MS2080A the IF filter has a nominal pass band of 40 MHz so the usable bandwidth for SB5 is nominally 40Mhz and SB1-4 are not present. For lower rates the usable bandwith listed is the nominal bandwith of the DDC anti aliasing filters.

Value Sample Rate(MHz) Bandwidth(MHz)

SB0 200 150

SB1 200 120

SB2 122.88 100

SB3 100 80

SB4 92.16 74

SB5 61.44 50

SB6 50 40

SB7 46.08 36

SB8 30.72 25

SB9 25 20

SB10 23.04 18

SB11 15.36 12

SB12 12.5 10

SB13 7.68 6

SB14 6.25 5

SB15 3.84 3

SB16 3.125 2.5

SB17 1.92 1.5

SB18 1.5625 1.25

SB19 0.36 0.28

SB20 0.045 0.036

Parameter(s):

<SB0|SB1|SB2|SB3|SB4|SB5|SB6|SB7|SB8|SB9|SB10|SB11|SB12|SB13|SB14|SB15|SB16|SB17|SB18|SB19|SB20>

Query Return:

SB0|SB1|SB2|SB3|SB4|SB5|SB6|SB7|SB8|SB9|SB10|SB11|SB12|SB13|SB14|SB15|SB16|SB17|SB18|SB19|SB20

Default Value: SB0

### [:SENSe]:IQ:SAMPle:CALibration:CONFiguration?

Title: IQ Sample Calibration Configuration

Description: In order to get the valid IQ correction factor, user should issue MEAS:IQ:CAPT command first. (Command only available in Spectrum and RTSA)

Returns a comma delimited list of I/Q measurement configuration and calibration coefficients information with 7 fields:

Center frequency,

Preamp state,

Input attenuation,

IQ base sample rate,

Decimation factor,

IQ capture bandwidth,

IQ correction factor in dB,

An invalid value of -20000 is returned if any of Center frequency, Preamp state, Input attenuation, IQ capture bandwidth settings is modified after issuing MEAS:IQ:CAPT command

To apply the IQ correction factor, please refer to the Remote Spectrum Monitoring System Help Introduction. Information is available in I/Q Capture Block Mode section of the manual (full path: Remote Spectrum Monitor/Programming with SCPI/I/Q Capture Block Mode)

### [:SENSe]:IQ:SAMPle:CONFiguration?

Title: IQ Configuration

Description: Returns a comma delimited list of I/Q measurement configuration information with 5 fields: Base sample rate, Number of bytes per frame, Effective bits per sample (see SENSE:IQ:BITS), Decimation factor, Timestamp (see SENSE:IQ:TIMESTAMP).

The base sample rate is generally fixed in a specific hardware revision; its value is provided here to enable calculation of the data rate, or time between samples, which is equal to the base sample rate divided by the decimation factor. The data rate is used when extrapolating embedded timestamps to other samples. For example, assuming a data rate of 1 MHz, if the nth sample contains an embedded timestamp of 1444753342s + 37531655ns, the n+1th sample would have an effective timestamp of 1444753342s + 37531655ns + (1/1MHz) = 1444753342s + 37531655ns + 1ms = 1444753342s + 37532655ns.

The number of samples per frame is calculated by: floor(bitsPerFrame / bitsPerSample), where bitsPerFrame = 8 * bytesPerFrame

Decimation factor is read only, and its value is derived from the current capture bandwith setting. There is a 1-to-1 correspondence between an IQ:BAND value and a decimation factor.

Number of bytes per sample is 8 for I/Q and 2 for raw ADC capture. The raw ADC capture must be parsed differently. See TRAC:IQ:DATA? for the data format.

This command provides the information necessary for clients to reconstruct I/Q data samples and timestamps from the raw data returned by TRAC:IQ:DATA?.

**[:SENSe]:IQ:SAVE:PROGress?**

Title: IQ Saving Progress

Description: This command returns a number between 0-100 that represents the percent completion of the task of saving the IQ data to a file.

Query Return: Percent

Default Value: 100

Range: 0% to 100%

**[:SENSe]:IQ:SIGNing <0 | 1 | ON | OFF>**
**[:SENSe]:IQ:SIGNing?**

Title: IQ Signing Status

Description: Toggle IQ signing On or Off. This command only is valid when option 0128 is installed and either IQ option 0124 or 0126.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:IQ:STATus?**

Title: IQ Capture Status

Description: This command returns current IQ Capture or IQ Streaming status.

Status description:

CONTinuous indicates continuous IQ capture is in progress.

SINGle indicates single IQ capture is in progress.

STREaming indicates IQ streaming is in progress.

READy indicates the instrument is ready for new IQ capture.

SAVing indicates that we are saving capture data into a file.

Query Return: CONT | SING | STRE | READ | SAV | WFIN | WFTR

Default Value: READy

**[:SENSe]:IQ:TIMEstamps <0 | 1 | ON | OFF>**
**[:SENSe]:IQ:TIMEstamps?**

Title: IQ Timestamps

Description: Enables or disables IQ timestamps.

When I/Q timestamps are enabled, timestamps will be embedded in the binary response data returned by TRACe:IQ:DATA?.

==========

MS2710xa:

The first 256 frames of each 1024 frame chunk use the least-significant bit of each I and Q sample in each frame for timestamping.

If SENS:IQ:BITS is 24 or 10, the timestamp does not reduce resolution.

If SENS:IQ:BITS is 16 or 8, the timestamp reduces the resolution by one bit for 256/1024 frames.

In the 16 bit case, 256/2048 samples are 15 bits. In the 8 bit case, 256/4096 samples are 7 bits.

If SENS:IQ:TIMESTAMPS is OFF, then the IQ data cannot be absolutely positioned in time, but all samples have full resolution.

This setting is ignored if SENSE:IQ:BITS is 24 or 10 bits because there are extra, otherwise unused bits.

The timestamps are shift-encoded in groups of 64 in bit 0, and there is a shift-encoded mark in bit 32.

==========

MS209xx:

Default is set to ON.

Frame chunks are composed of 64 frames of 64-bits in length. Each frame uses the least-significant bit of each I and Q sample for timestamping.

If SENS:IQ:BITS is I10, the timestamp does not reduce resolution.

If SENS:IQ:BITS is F32, I32, I16, A16, or I8, the timestamp reduces the resolution by one bit.

In the F32, I32 cases, each sample is 31 bits. In the I16 and A16 cases, 1/2 samples are 15 bits. In the I8 case, 1/4 samples are 7 bits. If SENS:IQ:TIMESTAMPS is OFF, then the IQ data cannot be absolutely positioned in time, but all samples have full resolution.

This setting is ignored if SENSE:IQ:BITS is 10 bits because there are extra, otherwise unused bits.

The timestamps are shift-encoded in groups of 64 in bit 64, and there is a shift-encoded mark in bit 32.

If GPS is not enabled, timestamps will use Internal PPS time. Enabling timestamps does not automatically enable GPS.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

### [:SENSe]:ISOTropic:AXIS <X|Y|Z|NONE>

Title: Isotropic Antenna Axis

Description: Sets the axis of the isotropic antenna. If the antenna is not present an error is put in the error queue.

Parameter(s): <X|Y|Z|NONE>

### [:SENSe]:ISOTropic:FACTors?

Title: Isotropic Antenna Factors

Description: Gets the antenna factors and other antenna metadata. If the antenna is not present an error is put in the error queue.

Example format:

#44624<ANRITSU>

<HEADER>

Rev-1

Isotropic H-Field Antenna

9 kHz - 300 MHz

2000-1800-R

1430101

07/22/2014

07/22/2014

</HEADER>

<ANTENNA_FACTORS>

0.009 70.18 70.18 70.18

0.010 68.36 68.36 68.36

0.011 66.61 66.61 66.61

0.012 65.34 65.34 65.34

...

280.000 51.81 51.81 51.81

290.000 52.87 52.87 52.87

300.000 53.13 53.13 53.13

</ANTENNA_FACTORS>

</ANRITSU>

**[:SENSe]:ISOTropic:INFOrmation?**

Title: Isotropic Antenna Information

Description: Gets the antenna metadata. The antenna information is same as the header section of :ISOTropic:FACTors.

Returns the information in json format.

{

"hardwareRevision":"Rev-1",

"antennaName":"Isotropic E-Field Antenna",

"frequencyRange":"700 MHz - 6 GHz",

"partNumber":"2000-1791-R",

"serialNumber":"14-17/116",

"manufacturingDate":"06/19/2014",

"lastCalibrationDate":"06/23/2014"

}

**[:SENSe]:LTE:CFI <AUTO|CFI1|CFI2|CFI3>**
**[:SENSe]:LTE:CFI?**

Title: CFI Selection

Description: Sets or queries the Control Format Indicator for the current LTE measurement.

Valid selections are: AUTO, 1, 2, or 3.

Parameter(s): <AUTO | CFI1 | CFI2 | CFI3>

Query Return: AUTO | CFI1 | CFI2 | CFI3

Default Value: AUTO

**[:SENSe]:LTE:OBANd**
**<INValid|UL1|DL1|UL2|DL2|UL3|DL3|UL4|DL4|UL5|DL5|UL6|DL6|UL7|DL7|UL8|D**
**L8|UL9|DL9|UL10|DL10|UL11|DL11|UL12|DL12|UL13|DL13|UL14|DL14|UL17|DL17**
**|UL18|DL18|UL19|DL19|UL20|DL20|UL21|DL21|UL22|DL22|UL23|DL23|UL24|DL24**
**|UL25|DL25|UL26|DL26|UL27|DL27|UL28|DL28|DL29|UL30|DL30|UL31|DL31|DL32**
**|UL33|DL33|UL34|DL34|UL35|DL35|UL36|DL36|UL37|DL37|UL38|DL38|UL39|DL39**
**|UL40|DL40|UL41|DL41|UL42|DL42|UL43|DL43|UL44|DL44|UL45|DL45|UL46|DL46**
**|UL47|DL47|UL48|DL48|UL49|DL49|UL50|DL50|UL51|DL51|UL52|DL52|UL53|DL53**
**|UL54|DL54|UL65|DL65|UL66|DL66|DL67|UL68|DL68|DL69|UL70|DL70|UL71|DL71**

**|UL72|DL72|UL73|DL73|UL74|DL74|DL75|DL76|UL85|DL85|UL87|DL87|UL88|DL88|U103|D103|U106|D106>**

**[:SENSe]:LTE:OBANd?**

Title: Operating Band Number

Description: Sets or queries the operating band number for the current demod measurement. Note that bands above 99 are written without the L to avoid truncation. For example, U103 instead of UL103.

Parameter(s):

<INValid|UL1|DL1|UL2|DL2|UL3|DL3|UL4|DL4|UL5|DL5|UL6|DL6|UL7|DL7|UL8|DL8|UL9|DL9|UL10|DL10|UL11|DL11|UL12|DL12|UL13|DL13|UL14|DL14|UL17|DL17|UL18|DL18|UL19|DL19|UL20|DL20|UL21|DL21|UL22|DL22|UL23|DL23|UL24|DL24|UL25|DL25|UL26|DL26|UL27|DL27|UL28|DL28|DL29|UL30|DL30|UL31|DL31|DL32|UL33|DL33|UL34|DL34|UL35|DL35|UL36|DL36|UL37|DL37|UL38|DL38|UL39|DL39|UL40|DL40|UL41|DL41|UL42|DL42|UL43|DL43|UL44|DL44|UL45|DL45|UL46|DL46|UL47|DL47|UL48|DL48|UL49|DL49|UL50|DL50|UL51|DL51|UL52|DL52|UL53|DL53|UL54|DL54|UL65|DL65|UL66|DL66|DL67|UL68|DL68|DL69|UL70|DL70|UL71|DL71|UL72|DL72|UL73|DL73|UL74|DL74|DL75|DL76|UL85|DL85|UL87|DL87|UL88|DL88|U103|D103|U106|D106>

Query Return:

INV|UL1|DL1|UL2|DL2|UL3|DL3|UL4|DL4|UL5|DL5|UL6|DL6|UL7|DL7|UL8|DL8|UL9|DL9|UL10|DL10|UL11|DL11|UL12|DL12|UL13|DL13|UL14|DL14|UL17|DL17|UL18|DL18|UL19|DL19|UL20|DL20|UL21|DL21|UL22|DL22|UL23|DL23|UL24|DL24|UL25|DL25|UL26|DL26|UL27|DL27|UL28|DL28|DL29|UL30|DL30|UL31|DL31|DL32|UL33|DL33|UL34|DL34|UL35|DL35|UL36|DL36|UL37|DL37|UL38|DL38|UL39|DL39|UL40|DL40|UL41|DL41|UL42|DL42|UL43|DL43|UL44|DL44|UL45|DL45|UL46|DL46|UL47|DL47|UL48|DL48|UL49|DL49|UL50|DL50|UL51|DL51|UL52|DL52|UL53|DL53|UL54|DL54|UL65|DL65|UL66|DL66|DL67|UL68|DL68|DL69|UL70|DL70|UL71|DL71|UL72|DL72|UL73|DL73|UL74|DL74|DL75|DL76|UL85|DL85|UL87|DL87|UL88|DL88|U103|D103|U106|D106

Default Value: INValid

**[:SENSe]:MIMO:ANTenna:PORT <AUTO|ANT0|ANT1|ANT2|ANT3>**

**[:SENSe]:MIMO:ANTenna:PORT?**

Title: Antenna Port

Description: Sets or queries the antenna port for the MIMO measurement.

Valid selections are: Auto (for autodetect), ANT0, ANT1, ANT2, and ANT3.

Parameter(s): <AUTO|ANT0|ANT1|ANT2|ANT3>

Query Return: AUTO|ANT0|ANT1|ANT2|ANT3

Default Value: AUTO

**[:SENSe]:MODE <SPECtrum|NRADio|LTE|PANanlyer|RTSA>**
**[:SENSe]:MODE?**

Title: Spa mode

Description: Set the operational mode of the Spa app.

Note that not all modes are available on all models.

SPECtrum for the default spectrum mode.

NRADio for 5G measurement mode.

RTSA for real time spectrum analysis mode.

LTE for 4G measurement mode.

EMFMeter for EMF meter measurement mode.

PANalyzer for Pulse Analyzer mode.

IASPectrum for Interference Analyzer Spectrum mode.

IARTsa for Interference Analyzer RTSA mode.

CS for Channel Scanner mode.

Parameter(s): <SPECtrum | NRADio | LTE | PANanlyer | RTSA>

Query Return: SPEC | NRAD | LTE | PAN | RTSA

Default Value: SPECtrum

**[:SENSe]:MODE:CATalog?**

Title: Spa Mode Catalog

Description: Returns a list of available modes for the Spa application. The response is a comma-separated list of mode names. See command [:SENSe]:MODE for the mode name specification.

**[:SENSe]:MODulation:REFerence?**

Title: Modulation Reference Point

Description: Queries the reference points for the current modulation type.

**[:SENSe]:MODulation:TYPE <ALL|QPSK|Q16|Q64|Q256>**
**[:SENSe]:MODulation:TYPE?**

Title: Modulation Type

Description: Sets or queries the modulation type for the current LTE or 5G measurement. Only QPSK is available in 5G measurement.

Valid selections are: ALL, QPSK, Q16 (16-QAM), Q64 (64-QAM), and Q256 (256-QAM).

Parameter(s): <ALL | QPSK | Q16 | Q64 | Q256>

Query Return: ALL | QPSK | Q16 | Q64 | Q256

Default Value: ALL

**[:SENSe]:NGFactor <ONESixth|HALF|ONE|TWO>**
**[:SENSe]:NGFactor?**

Title: Ng Factor

Description: Sets or queries the Ng factor for the number of PHICH groups calculation.

Apply to TDD duplex type only.

Valid selections are: ONESixth (1/6), HALF (1/2), ONE (1), and TWO (2).

Parameter(s): <ONESixth|HALF|ONE|TWO>

Query Return: ONES|HALF|ONE|TWO

Default Value: ONESixth

**[:SENSe]:NRADio:MAPPing:PATTern <P1|P2|AUTO>**
**[:SENSe]:NRADio:MAPPing:PATTern?**

Title: Mapping Pattern

Description: Sets or queries the Mapping Pattern used in the New Radio mode.

Available selections are: P1 (Mapping Pattern 1), or P2 (Mapping Pattern 2)

Parameter(s): <P1|P2|AUTO>

Query Return: P1|P2|AUTO

Default Value: AUTO

**[:SENSe]:OBANd**
**<INValid|GLOBal|N1|N2|N3|N5|N7|N8|N12|N13|N14|N18|N20|N24|N25|N26|N28|N29|N30|N34|N38|N39|N40|N41|N46|N48|N50|N51|N53|N54|N65|N66|N67|N70|N71|N74|N75|N76|N77|N78|N79|N85|N90|N91|N92|N93|N94|N96|N100|N101|N102|N104|N105|N257|N258|N259|N260|N261|N262|N263>**
**[:SENSe]:OBANd?**

Title: Operating Band Number

Description: Sets or queries the operating band number for the current demod measurement. Sets via this command will also restrict available ARF and GS Channel numbers.

Band availability may be limited by options purchased.

When GLOBal is selected, applies the global ARFCN to GSCN calculation only.

When Nx is selected, applies any band restrictions over the global ARFCN to GSCN calculation.

Parameter(s):

<INValid|GLOBal|N1|N2|N3|N5|N7|N8|N12|N13|N14|N18|N20|N24|N25|N26|N28|N29|N30|N34|N38|N39|N40|N41|N46|N48|N50|N51|N53|N54|N65|N66|N67|N70|N71|N74|N75|N76|N77|N78|N79|N85|N90|N91|N92|N93|N94|N96|N100|N101|N102|N104|N105|N257|N258|N259|N260|N261|N262|N263>

Query Return:

INV|GLOB|N1|N2|N3|N5|N7|N8|N12|N13|N14|N18|N20|N24|N25|N26|N28|N29|N30|N34|N38|N39|N40|N41|N46|N48|N50|N51|N53|N54|N65|N66|N67|N70|N71|N74|N75|N76|N77|N78|N79|N85|N90|N91|N92|N93|N94|N96|N100|N101|N102|N104|N105|N257|N258|N259|N260|N261|N262|N263

Default Value: INValid

**[:SENSe]:OBWidth:METHod <XDB|PERCent>**
**[:SENSe]:OBWidth:METHod?**

Title: Occupied Bandwidth Method

Description: Sets the method for calculating occupied bandwidth. XDB calculates the occupied bandwidth based on points a specified number of dB below the carrier. Issue command [:SENSe]:OBWidth:XDB to set the number of dB to be used. PERCent calculates the occupied bandwidth based on points a specified percentage of the carrier power below the carrier. Issue command [:SENSe]:OBWidth:PERCent to set the percentage to be used.

The measurement always gives a result, even if there is no signal. For example, the 100 dBc Occupied Bandwidth is the current full span.

Parameter(s): <XDB|PERCent>

Query Return: XDB|PERC

Default Value: PERCent

**[:SENSe]:OBWidth:PERCent <numeric_value>**
**[:SENSe]:OBWidth:PERCent?**

Title: Occupied Bandwidth Percent

Description: This command sets the percentage of carrier power used to measure the occupied bandwidth. This value is used in the measurement if :SENSe:OBWidth:METHod is set to PERCent.

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 99

Range: 1e-06 to 99.999999

**[:SENSe]:OBWidth:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:OBWidth:STATe?**

Title: Occupied Bandwidth State

Description: Sets the state of the occupied bandwidth measurement, ON or OFF. When using :CONFigure:OBWidth, the state is automaticaly set to ON

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:OBWidth:XDB <numeric_value> {DB}**
**[:SENSe]:OBWidth:XDB?**

Title: Occupied Bandwidth XDB

Description: This command sets the number of dB below the carrier used to measure the occupied bandwidth. This value is used in the measurement if :SENSe:OBWidth:METHod is set to XDB.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {DB}

Query Return: Numeric (dB)

Default Value: 3 dB

Default Unit: dB

Range: 0.001 dB to 100 dB

**[:SENSe][:OTA]:MAPPing:STATe <0 | 1 | ON | OFF>**
**[:SENSe][:OTA]:MAPPing:STATe?**

Title: OTA Coverage Mapping State

Description: Enable or disable OTA Coverage Mapping measurement

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:PANalyzer?**

Title: Get Pulse Analyzer Measurement

Description: Return the current Pulse Analyzer measurement data. Unlike :FETCh:PANalyzer this command does not wait for the current measurement to complete before returning results.

**[:SENSe]:PANalyzer:ALL?**

Title: Get All Pulse Analyzer Data

Description: Return all trace data as well as the latest Pulse Analyzer measurement data, returned as a block of binary trace data followed by the JSON PAN results. Unlike :FETCh:PANalyzer this command does not wait for the current measurement to complete before returning results.

**[:SENSe]:PCI?**

Title: Get PCI Measurements

Description: Returns the last stored PCI measurement results. This command returns cell id, group id, corrected carrier frequency, detected beam,

frame start, frequency error, sector id, group id, and time offset(in ns) as output in json format.

This command does not wait for the current measurement to complete before returning results.

An example is as follows:

{
"carrierId":1,
"cellId": 0,
"cellIdGroup": 0,
"correctedCarrierFrequency": 4499999985.264854,
"detectedBeam": 4,
"frameStart": 233097,
"frequencyError": -14.735145792134368,
"ppmFrequencyError": 0.09346635883256266,
"sectorId": 0,
"status": {
"hardwareErrors": {
"ADCOverRange": 0,
"DeviceAccess": 0,
"EEPROM": 0,
"IrregularData": 0,
"Limit": 0,
"LockLO1": 0,
"LockLO2": 0,
"NoCal": 0,
"Overpower": 0,
"SLOLock": 0,
"Saturation": 0,
"TgLockLo1": 0
},
"statusMessage": "statusPciDone",
"statusNumber": 10
},
"gps":{
"coordinates":"37.14681, -121.65609, 128m",
"time":"Fri Jun 28 19:56:45 2019 GMT"
},

"timeOffset": -634782.9687499882

}

**[:SENSe]:POWer:RF:ATTenuation <numeric_value> {DB}**
**[:SENSe]:POWer:RF:ATTenuation?**

Title: Input Attenuation

Description: Sets the input attenuation. Note that issuing this command will set the automatic input attenuation OFF. Actual minimum and maximum values vary, depending on model and options.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {DB}

Query Return: Numeric (dB)

Default Value: 30 dB

Default Unit: dB

Range: 0 dB to 65 dB

**[:SENSe]:POWer:RF:ATTenuation <numeric_value> {DB}**
**[:SENSe]:POWer:RF:ATTenuation?**

Title: Input Attenuation

Description: Sets the input attenuation. Note that issuing this command will set the automatic input attenuation OFF. Actual minimum and maximum values vary, depending on model and options.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {DB}

Query Return: Numeric (dB)

Default Value: 30 dB

Default Unit: dB

Range: 0 dB to 65 dB

**[:SENSe]:POWer:RF:ATTenuation:AUTO <0 | 1 | ON | OFF>**
**[:SENSe]:POWer:RF:ATTenuation:AUTO?**

Title: RF Attenuation Auto

Description: Sets the input attenuation coupling. Setting the value to ON or 1 will result in the input attenuation being coupled to the reference level. Setting the value to OFF or 0 will result in the input attenuation being uncoupled from the reference level. That is, changing the reference level will not change the input attenuation. When this command is issued, the input attenuator setting itself will not change. The default value is ON. That is, sending :SENS:POW:ATT:AUTO is equivalent to sending :SENS:POW:ATT:AUTO ON.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:POWer:RF:ATTenuation:AUTO <0 | 1 | ON | OFF>**
**[:SENSe]:POWer:RF:ATTenuation:AUTO?**

Title: RF Attenuation Auto

Description: Sets the input attenuation coupling. Setting the value to ON or 1 will result in the input attenuation being coupled to the reference level. Setting the value to OFF or 0 will result in the input attenuation being uncoupled from the reference level. That is, changing the reference level will not change the input attenuation. When this command is issued, the input attenuator setting itself will not change. The default value is ON. That is, sending :SENS:POW:ATT:AUTO is equivalent to sending :SENS:POW:ATT:AUTO ON.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:POWer:RF:GAIN:AUTO <0 | 1 | ON | OFF>**
**[:SENSe]:POWer:RF:GAIN:AUTO?**

Title: Auto Gain Ranging

Description: Sets the state of the auto gain-ranging functionality.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:POWer:RF:GAIN:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:POWer:RF:GAIN:STATe?**

Title: Preamp

Description: Sets the state of the preamp. Note that this may cause a change in the reference level and/or attenuation.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:POWer:RF:GAIN:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:POWer:RF:GAIN:STATe?**

Title: Preamp

Description: Sets the state of the preamp. Note that this may cause a change in the reference level and/or attenuation.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:PULSe:LABel:ALL:OFF**

Title: Pulse Label All Off

Description: Turn off all pulse labels.

**[:SENSe]:PULSe:LABel:ALL:ON**

Title: Pulse Label All On

Description: Turn on all pulse labels.

**[:SENSe]:PULSe:LABel:INSTant:CENTer <0 | 1 | ON | OFF>**
**[:SENSe]:PULSe:LABel:INSTant:CENTer?**

Title: Pulse Instant Label Center

Description: Sets/queries the state to enable pulse center instant marker and label.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:PULSe:LABel:INSTant:REFerence:DURation:FIRSt <0 | 1 | ON | OFF>**
**[:SENSe]:PULSe:LABel:INSTant:REFerence:DURation:FIRSt?**

Title: Pulse Instant Label Reference Duration First Transition

Description: Sets/queries the state to enable pulse reference duration first transition instant marker and label.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:PULSe:LABel:INSTant:REFerence:DURation:SECond <0 | 1 | ON | OFF>**
**[:SENSe]:PULSe:LABel:INSTant:REFerence:DURation:SECond?**

Title: Pulse Instant Label Reference Duration Second Transition

Description: Sets/queries the state to enable pulse reference duration second transition instant marker and label.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:PULSe:LABel:INSTant:REFerence:HIGH:FIRSt <0 | 1 | ON | OFF>**
**[:SENSe]:PULSe:LABel:INSTant:REFerence:HIGH:FIRSt?**

Title: Pulse Instant Label Reference High First Transition

Description: Sets/queries the state to enable pulse reference high first transition instant marker and label.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:PULSe:LABel:INSTant:REFerence:HIGH:SECond <0 | 1 | ON | OFF>**
**[:SENSe]:PULSe:LABel:INSTant:REFerence:HIGH:SECond?**

Title: Pulse Instant Label Reference High Second Transition

Description: Sets/queries the state to enable pulse reference high second transition instant marker and label.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:PULSe:LABel:INSTant:REFerence:LOW:FIRSt <0 | 1 | ON | OFF>**
**[:SENSe]:PULSe:LABel:INSTant:REFerence:LOW:FIRSt?**

Title: Pulse Instant Label Reference Low First Transition

Description: Sets/queries the state to enable pulse reference low first transition instant marker and label.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:PULSe:LABel:INSTant:REFerence:LOW:SECond <0 | 1 | ON | OFF>**
**[:SENSe]:PULSe:LABel:INSTant:REFerence:LOW:SECond?**

Title: Pulse Instant Label Reference Low Second Transition

Description: Sets/queries the state to enable pulse reference low second transition instant marker and label.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:PULSe:LABel:INSTant:TRANsition:POST:ABERration <0 | 1 | ON | OFF>**
**[:SENSe]:PULSe:LABel:INSTant:TRANsition:POST:ABERration?**

Title: Pulse Instant Label Post-transition Aberration

Description: Sets/queries the state to enable pulse post-transition aberration region instant marker and label.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:PULSe:LABel:INSTant:TRANsition:POST:OVER <0 | 1 | ON | OFF>**
**[:SENSe]:PULSe:LABel:INSTant:TRANsition:POST:OVER?**

Title: Pulse Instant Label Post-transition Overshoot

Description: Sets/queries the state to enable pulse post-transition overshoot region instant marker and label.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:PULSe:LABel:INSTant:TRANsition:POST:UNDer <0 | 1 | ON | OFF>**
**[:SENSe]:PULSe:LABel:INSTant:TRANsition:POST:UNDer?**

Title: Pulse Instant Label Post-transition Undershoot

Description: Sets/queries the state to enable pulse post-transition undershoot region instant marker and label.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:PULSe:LABel:INSTant:TRANsition:PRE:ABERration <0 | 1 | ON | OFF>**
**[:SENSe]:PULSe:LABel:INSTant:TRANsition:PRE:ABERration?**

Title: Pulse Instant Label Pre-transition Aberration

Description: Sets/queries the state to enable pulse pre-transition aberration region instant marker and label.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:PULSe:LABel:INSTant:TRANsition:PRE:OVER <0 | 1 | ON | OFF>**
**[:SENSe]:PULSe:LABel:INSTant:TRANsition:PRE:OVER?**

Title: Pulse Instant Label Pre-transition Overshoot

Description: Sets/queries the state to enable pulse pre-transition overshoot region instant marker and label.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:PULSe:LABel:INSTant:TRANsition:PRE:UNDer <0 | 1 | ON | OFF>**
**[:SENSe]:PULSe:LABel:INSTant:TRANsition:PRE:UNDer?**

Title: Pulse Instant Label Pre-transition Undershoot

Description: Sets/queries the state to enable pulse pre-transition undershoot region instant marker and label.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:PULSe:LABel:LEVel:REFerence:DURation <0 | 1 | ON | OFF>**
**[:SENSe]:PULSe:LABel:LEVel:REFerence:DURation?**

Title: Pulse Level Label Reference Duration

Description: Sets/queries the state to enable pulse reference level duration marker and label.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:PULSe:LABel:LEVel:REFerence:HIGH <0 | 1 | ON | OFF>**
**[:SENSe]:PULSe:LABel:LEVel:REFerence:HIGH?**

Title: Pulse Level Label Reference High

Description: Sets/queries the state to enable pulse reference level high marker and label.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:PULSe:LABel:LEVel:REFerence:LOW <0 | 1 | ON | OFF>**
**[:SENSe]:PULSe:LABel:LEVel:REFerence:LOW?**

Title: Pulse Level Label Reference Low

Description: Sets/queries the state to enable pulse reference level low marker and label.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:PULSe:LABel:LEVel:STATe:HIGH <0 | 1 | ON | OFF>**
**[:SENSe]:PULSe:LABel:LEVel:STATe:HIGH?**

Title: Pulse Level Label State High

Description: Sets/queries the state to enable pulse state level high marker and label.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:PULSe:LABel:LEVel:STATe:LOW <0 | 1 | ON | OFF>**
**[:SENSe]:PULSe:LABel:LEVel:STATe:LOW?**

Title: Pulse Level Label State Low

Description: Sets/queries the state to enable pulse state level low marker and label.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:PULSe:LEVel:TYPE <AUTO|USER>**
**[:SENSe]:PULSe:LEVel:TYPE?**

Title: Pulse Level Type

Description: Sets/queries whether the system will use measurement data to calculate the pulse's top and bottom (AUTO), or use user-supplied values (USER).

Parameter(s): <AUTO|USER>

Query Return: AUTO|USER

Default Value: AUTO

**[:SENSe]:PULSe:LEVel:USER:BOTTom <numeric_value> {<amplitude_units>}**
**[:SENSe]:PULSe:LEVel:USER:BOTTom?**

Title: Pulse User Level Bottom

Description: Sets/queries the amplitude level to be used to define the bottom of the pulse.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {<amplitude_units>}

Query Return: Numeric (<amplitude_units>)

Default Value: -100 dBm

Default Unit: dBm

Range: -100 dBm to 100 dBm

**[:SENSe]:PULSe:LEVel:USER:TOP <numeric_value> {<amplitude_units>}**
**[:SENSe]:PULSe:LEVel:USER:TOP?**

Title: Pulse User Level Top

Description: Sets/queries the amplitude level to be used to define the top of the pulse.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {<amplitude_units>}

Query Return: Numeric (<amplitude_units>)

Default Value: 0 dBm

Default Unit: dBm

Range: -100 dBm to 100 dBm

**[:SENSe]:PULSe:REFerence:DURation <percent_value>**
**[:SENSe]:PULSe:REFerence:DURation?**

Title: Pulse Duration Reference
Description: Sets/queries the percentage value to be used to define the pulse duration.
Parameter(s): <percent_value>
Query Return: Percent
Default Value: 50
Range: 0.2% to 99.9%

**[:SENSe]:PULSe:REFerence:HIGH <percent_value>**
**[:SENSe]:PULSe:REFerence:HIGH?**

Title: Pulse Reference High
Description: Sets/queries the percentage value to be used to define a high state.
Parameter(s): <percent_value>
Query Return: Percent
Default Value: 90
Range: 0.3% to 100%

**[:SENSe]:PULSe:REFerence:LOW <percent_value>**
**[:SENSe]:PULSe:REFerence:LOW?**

Title: Pulse Reference Low
Description: Sets/queries the percentage value to be used to define a low state.
Parameter(s): <percent_value>
Query Return: Percent
Default Value: 10
Range: 0.1% to 99.8%

**[:SENSe]:PULSe:SIMulation:AMPLitude:HIGH <numeric_value>**
**{<amplitude_units>}**
**[:SENSe]:PULSe:SIMulation:AMPLitude:HIGH?**

Title: Pulse Simulation High State
Description: Sets/queries the high state amplitude of the simulated pulse signal.
See UNIT:POWer to query or set the current amplitude units.
Parameter(s): <numeric_value> {<amplitude_units>}
Query Return: Numeric (<amplitude_units>)
Default Value: 2 dBm
Default Unit: dBm
Range: -200 dBm to 200 dBm

**[:SENSe]:PULSe:SIMulation:AMPLitude:LOW <numeric_value> {<amplitude_units>}**
**[:SENSe]:PULSe:SIMulation:AMPLitude:LOW?**

Title: Pulse Simulation Low State

Description: Sets/queries the low state amplitude of the simulated pulse signal.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {<amplitude_units>}

Query Return: Numeric (<amplitude_units>)

Default Value: -33 dBm

Default Unit: dBm

Range: -200 dBm to 200 dBm

**[:SENSe]:PULSe:SIMulation:DUTY <numeric_value>**
**[:SENSe]:PULSe:SIMulation:DUTY?**

Title: Pulse Simulation Duty Factor

Description: Sets/queries the duty factor of the simulated pulse signal.

This setting apply to TRAin setting value of PULSe:SIMulation:WAVeform only.

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 0.5

Range: 0.01 to 1

**[:SENSe]:PULSe:SIMulation:PERiod <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**[:SENSe]:PULSe:SIMulation:PERiod?**

Title: Pulse Simulation Period

Description: Sets/queries the period of the simulated pulse signal.

This setting apply to TRAin setting value of PULSe:SIMulation:WAVeform only.

Due to display point count limitation, displaying a pulse train with more than 333 pulses on GUI is not possible, displaying a flat line at high state amplitude instead.

Number of pulses in pulse train can be calculated as Sweep Time / Pulse Simulation Period.

For example, when sweep time is set to 1 s, GUI will display a flat line when pulse simulation is less tham 3 ms.

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 250 ms

Default Unit: ms

Range: 0.02 ms to 60000 ms

**[:SENSe]:PULSe:SIMulation:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:PULSe:SIMulation:STATe?**

Title: Pulse Simulation State

Description: Sets/queries the state of pulse simulation mode.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:PULSe:SIMulation:WAVeform <SPOStive|SNEGative|TRAin|DOUBle>**
**[:SENSe]:PULSe:SIMulation:WAVeform?**

Title: Pulse Simulation Waveform Type

Description: Sets/queries the waveform type of the simulated pulse signal.

SPOStive for single positive pulse.

SNEGative for single negative pulse.

TRAin for pulse train.

DOUBle for double pulse.

PULSe:SIMulation:PERiod and PULSe:SIMulation:DUTY setting apply to TRAin type only.

Parameter(s): <SPOStive|SNEGative|TRAin|DOUBle>

Query Return: SPOS|SNEG|TRA|DOUB

Default Value: TRAin

**[:SENSe]:PULSe:TYPE <POSitive|NEGative>**
**[:SENSe]:PULSe:TYPE?**

Title: Pulse Type

Description: Sets/queries pulse type to guide Pulse Analyzer measurement.

Parameter(s): <POSitive|NEGative>

Query Return: POS|NEG

Default Value: POSitive

**[:SENSe]:SCANner?**

Title: Get Cell Scanner

Description: Returns the cell scanner measurements in a json format. unlike [:SENSe]:FETCH:SCANner this command does not block and will return the previous result. Returned data format varies based on whether the current mode is 5GNR or LTE.

5GNR Mode format:


{

"carrierId":1,

"gps":{

"coordinates":"37.14681, -121.65609, 128m",

"time":"Fri Jun 28 19:56:45 2019 GMT"

},

"status":{

"hardwareErrors":{

"ADCOverRange":0,

"DeviceAccess":0,

"EEPROM":0,

"IrregularData":0,

"Limit":0,

"LockLO1":0,

"LockLO2":0,

"NoCal":0,

"Overpower":0,

"SLOLock":0,

"Saturation":0,

"TgLockLo1":0

},

"statusMessage":"statusCellScannerDone",

"statusNumber":13

},

"values":[

{

"cellId":1007,

"cellIdGroup":335,

"frameStart":1920521,

"sectorId":0,

"timeOffset":-3490194.091145895,

"values":[

{

"beamIndex":0,

"beamPresent":1,
"ssRmsEvm":6.488177299499512,
"ssRsrp":-67.18608535715634,
"ssRsrq":-9.913812153567978,
"ssSinr":23.757546218871823
},
...
{
"beamIndex":7,
"beamPresent":1,
"ssRmsEvm":6.488177299499512,
"ssRsrp":-84.22420720274197,
"ssRsrq":null,
"ssSinr":null
}
]
}
]
}

LTE Mode format:

{
"carrierId":1,
"gps":{
"coordinates":"nan, nan,nanm","time":""
},
"status":{
"hardwareErrors":{
"ADCOverRange":0,
"DeviceAccess":0,
"EEPROM":0,
"IrregularData":0,
"Limit":0,
"LockLO1":0,
"LockLO2":0,
"NoCal":0,
"Overpower":0,
"SLOLock":0,
"Saturation":0,
"TgLockLo1":0
},

```
"statusMessage":"statusCellScannerDone",
"statusNumber":13
},
"values":[
{
"cellId":2,
"cellIdGroup":0,
"crsRsrp":-68.46522521972656,
"crsRsrq":0.2733601927757263,
"crsSinr":34.74685287475586,
"dominance":null,
"dominantCell":true,
"frameStart":-1,
"frequencyError":227.3341827392578,
"pbch":{
"peakEvm":{
"percent":4.949735641479492,
"subcarrierNumber":-1,
"symbolNumber":-1
},
"rmsEvm":2.2826967239379883
},
"ppmFrequencyError":0.06495262363978795,
"sectorId":2,
"ssRsrp":-68.43709564208984
}
]
}
```


**[:SENSe]:SCANner:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:SCANner:STATe?**

Title: Demod Scanner State Command
Description: Turns on cell scanner, disables other measurements
Parameter(s): <0 | 1 | ON | OFF>
Query Return: 0 | 1
Default Value: OFF

**[:SENSe]:SEMask:CENTer:BWIDth <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe]:SEMask:CENTer:BWIDth?**

Title: Center Channel Bandwidth

Description: Set or query the bandwidth of the central channel of the Spectral Emission Mask measurement. The center channel is used to calculate the reference power when reference power calculation is set to AUTO.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

Query Return: Numeric (Hz)

Default Value: 100000000 Hz

Default Unit: Hz

Range: 10000 Hz to 1000000000 Hz

**[:SENSe]:SEMask:MASK**
**<GT38|LT38|W43|W41|W35|W31|WK43|WK41|WK35|WK31|LK24|LK22|LK20|NK38|NK35|NK28|NK24|CUSTom>**
**[:SENSe]:SEMask:MASK?**

Title: Spectral Mask

Description: Sets the preprogrammed spectral mask to be use for the Spectral Emission Mask measurement. Available selections are specified in the following table: Spectral Mask Description GT38 5G NR P > 38 dBm LT38 5G NR P < 38 dBm W43 WCDMA P >= 43 dBm W41 WCDMA 39 <= P < 43 dBm W35 WCDMA 31 <= P < 39 dBm W31 WCDMA P < 31 dBm WK43 WCDMA KR P >= 43 dBm WK41 WCDMA KR 39 <= P < 43 dBm WK35 WCDMA KR 31 <= P < 39 dBm WK31 WCDMA KR P < 31 dBm LK24 LTE KR P >= 24 dBm LK22 LTE KR 20 <= P < 24 dBm LK20 LTE KR P < 20 dBm NK38 5G NR KR P >= 38 dBm NK35 5G NR KR 31 <= P < 38 dBm NK28 5G NR KR 24 <= P < 31 dBm NK24 5G NR KR P < 24 dBm CUSTom User Defined Mask from [:SENSe]:SEMask:MASK:CUSTom\

Parameter(s):

<GT38|LT38|W43|W41|W35|W31|WK43|WK41|WK35|WK31|LK24|LK22|LK20|NK38|NK35|NK28|NK24|CUSTom>

Query Return:

GT38|LT38|W43|W41|W35|W31|WK43|WK41|WK35|WK31|LK24|LK22|LK20|NK38|NK35|NK28|NK24|CUST

Default Value: GT38

**[:SENSe]:SEMask:MASK:CUSTom <string>,<block data>**
**[:SENSe]:SEMask:MASK:CUSTom?**

Title: Custom Spectral Emission Mask Segment Data

Description: This command takes String as a name and imports file contents to the instrument. Data is transferred to the instrument as an IEEE definite length arbitrary block response, which has the form <header><block>.

Sting Value: The name value sent

Block Data: The data to be written to the instrument in block data format

Please note that the maximum file transfer size to the instrument is 25 MB. The ASCII header specifies the number of data bytes of the file. It looks like #AX, where A is the number of digits in X and X is the number of bytes in the <block>. The first character is the literal ascii hash '#' 043. The second character (A) is a single ascii digit '1' to '9' describing the number of bytes in the length section (X). This number is called nlength. The next nlength bytes make up an ascii string of digits '1' to '9' describing the length of the <block> data.

For example, if the first 6 bytes are #49999, then the nlength is 4. The 4 bytes of length are 9999. After that follows the <block>, which would be 9999 bytes in size.

The contents of the block data should format to csv with each line being a segment in the custom SEM.

Needed values for a segment are: startFrequency, stopFrequency, rbw, vbw, isStartAmpRelative, isStopAmpRelative, startAmplitude, stopAmplitude Start and Stop Frequencies are relative to the center frequency.

Example of a properly formatted mask: -4.00e7,-1.01e7,1.00e6,1.00e5,0,0,-15,-15 1.01e7,4.00e7,1.00e6,1.00e5,0,0,-15,-15

Query returns a String formatted as Json with name: and mask: as the keys

example:

{ name: maskname, mask: -4.00e7,-1.01e7,1.00e6,1.00e5,0,0,-15,-15

1.01e7,4.00e7,1.00e6,1.00e5,0,0,-15,-15 }\

Parameter(s): <string>,<block data>

Default Value: {"name":"example.csv"}

**[:SENSe]:SEMask:MASK:TYPe?**

Title:

Description: Query the Spectral Emission Mask offest type of [:SENSe]:SEMask:MASK selection. Type Description CHANnel Mask segments offset from the edges of channel bandwidth. CENTer Mask segments offset from the center frequency.\

Query Return: CHAN | CENT

Default Value: CHANnel

**[:SENSe]:SEMask:RPOWer <numeric_value> {<amplitude_units>}**
**[:SENSe]:SEMask:RPOWer?**

Title: Manual Reference Power

Description: Set or query the manual reference power value used in the Spectral Emission Mask measurement when reference power calculation method ([SENSe]:SEMask:RPOWer:STATe) is set to OFF or 0. When the reference power calculation method is set to ON or 1, this manual reference power value will be unused.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {<amplitude_units>}

Query Return: Numeric (<amplitude_units>)

Default Value: 0 dBm

Default Unit: dBm

Range: -200 dBm to 200 dBm

**[:SENSe]:SEMask:RPOWer:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:SEMask:RPOWer:STATe?**

Title: Reference Power Calculation

Description: Set or query the method for how the reference power is calculated in the Spectral Emission Mask measurement. Setting the value to ON or 1, the reference power will be obtained automatically by measuring the channel power of the center channel bandwidth. Setting the value to OFF or 0, the reference power will be the value set using the manual reference power command ([:SENSe]:SEMask:RPOWer).

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:SEMask:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:SEMask:STATe?**

Title: Spectral Emission Mask State

Description: Sets the state of the Spectral Emission Mask measurement, ON or OFF. When using :CONFigure:SEMask,the state is automatically set to ON.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:SIB:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:SIB:STATe?**

Title: SIB1 Measurement State Command

Description: Sets or queries SIB1 measurement state

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:SSB?**

Title: Get SSB Measurements

Description: Returns the last stored SSB measurement results.

This command returns the beam numbers with the best EVM and power, followed by Power and EVM results

for PSS, SSS, PBCH, PBCH DM-RS and SS Signal Quality Results in a json format.

This command does not wait for the current measurement to complete before returning results.

eg:


{

"bestEvmBeam":5,

"bestPowerBeam":0,

"status": {

"hardwareErrors": {

"ADCOverRange": 0,

"DeviceAccess": 0,

"EEPROM": 0,

"IrregularData": 0,

"Limit": 0,

"LockLO1": 0,

"LockLO2": 0,

"NoCal": 0,

"Overpower": 0,

"SLOLock": 0,

"Saturation": 0,

"TgLockLo1": 0

},

"statusMessage": "statusPciDone",

"statusNumber": 10

},

"gps":{

"coordinates":"37.14681, -121.65609, 128m",

"time":"Fri Jun 28 19:56:45 2019 GMT"

},

"values":[

{

"dmRsPbch":{

"peakEvm":{

"percent":11.926020622253418,

"subcarrierNumber":1529,

```
"symbolNumber":3
},
"power":-31.813847928749347,
"rmsEvm":3.813983917236328
},
"pbch":{
"peakEvm":{
"percent":13.430730819702148,
"subcarrierNumber":1638,
"symbolNumber":5
},
"power":-31.8139914869584,
"rmsEvm":3.736370325088501
},
"pss":{
"peakEvm":{
"percent":12.912864685058594,
"subcarrierNumber":1638,
"symbolNumber":2
},
"power":-31.704519932556128,
"rmsEvm":1.7515575885772705
},
"ssRsrp":-31.750750515723816,
"ssRsrq":-10.728772378478801,
"ssSinr":-13.878087163543444,
"sss":{
"peakEvm":{
"percent":11.369827270507812,
"subcarrierNumber":1638,
"symbolNumber":4
},
"power":-31.750750515723816,
"rmsEvm":2.5253000259399414
}
},
{ ... },
{ ... },
{ ... },
{ ... },
```

{ ... },
{ ... },
{ ... }
]
}

**[:SENSe]:SSB:FREQuency <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe]:SSB:FREQuency?**

Title: SSB Frequency
Description: Sets or queries the SS Block center frequency in Hz.
Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}
Query Return: Numeric (Hz)
Default Value: 0 Hz
Default Unit: Hz
Range: -100000000000 Hz to 300000000000 Hz

**[:SENSe]:SSB:OFFSet <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe]:SSB:OFFSet?**

Title: SSB Frequency Offset
Description: Sets or queries the SS Block frequency offset from the current center frequency in Hz.
Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}
Query Return: Numeric (Hz)
Default Value: 0 Hz
Default Unit: Hz
Range: -200000000000 Hz to 200000000000 Hz

**[:SENSe]:SSB:OFFSet:AUTO:ABORt**

Title: SSB Autodetection Abort
Description: Aborts the current SSB Autodetection operation. System may take several seconds before the current operation terminates completely. The [:SENSe]:SSB:OFFSet:AUTO:STATus query

will return "ABOR" when complete.

**[:SENSe]:SSB:OFFSet:AUTO:STARt**

Title: SSB Autodetection Start
Description: Starts the SSB Autodetection operation.

**[:SENSe]:SSB:OFFSet:AUTO:STATus?**

Title: SSB Autodetection Status

Description: Queries the current status of the SSB Autodetection operation. Valid values are:

INACtive: Operation has not been performed with the current set of parameters. ACTive: Operation is in progress. EXTended: Extensive search is in progress. PASS: Operation has completed successfully, and the new SSB Offset has been applied. FAIL: Operation has completed, but did not find a valid SSB. ABORted: Operation was aborted before completion.

Query Return: INAC | ACT | EXT | PASS | FAIL | ABOR

Default Value: INACtive

**[:SENSe][:SSB]:SCSPacing <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe][:SSB]:SCSPacing?**

Title: SSB Subcarrier Spacing

Description: Sets or queries the SS Block subcarrier spacing for the current demod measurement.

Full set of possible values are 15, 30, 120, 240 kHz.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

Query Return: Numeric (Hz)

Default Value: 30000 Hz

Default Unit: Hz

Range: 15000 Hz to 240000 Hz

**[:SENSe][:SSB]:SCSPacing:CATalog?**

Title: SSB Subcarrier Spacing Catalog

Description: Queries a list of all valid SS block subcarrier spacing values based on current settings.

Data is returned in a comma-separated list of frequencies in Hz, e.g. "15000,30000,120000,240000"

Default Value: 15000,30000,120000,240000

**[:SENSe]:STATistics:CLEar**

Title: Clear measurement statistics

Description: Clears measurement statistics, (E.G. mean frequency error)

**[:SENSe]:SUBFrame <numeric_value>**
**[:SENSe]:SUBFrame?**

Title: Subframe Number

Description: Sets or queries the subframe number for the current LTE or 5G measurement.

Values range from 0-9.

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 0

Range: 0 to 9

**[:SENSe]:SUBFrame:CONFiguration**
**<AUTO|SSF0|SSF1|SSF2|SSF3|SSF4|SSF5|SSF6|SSF7|SSF8|SSF9|INValid>**
**[:SENSe]:SUBFrame:CONFiguration?**

Title: Special Subframe Configuration

Description: Sets or queries the special subframe configuration for the current LTE measurement.

Parameter(s): <AUTO | SSF0 | SSF1 | SSF2 | SSF3 | SSF4 | SSF5 | SSF6 | SSF7 | SSF8 | SSF9 | INValid>

Query Return: AUTO | SSF0 | SSF1 | SSF2 | SSF3 | SSF4 | SSF5 | SSF6 | SSF7 | SSF8 | SSF9 | INV

Default Value: SSF0

**[:SENSe]:SUBFrame:SLOT <numeric_value>**
**[:SENSe]:SUBFrame:SLOT?**

Title: Slot Number

Description: Sets or queries the slot number of the current selected subframe in the LTE or 5G frame power measurement.

Allowed values depend on Subcarrier Spacing and Periodicity.

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 0

Range: 0 to 1

**[:SENSe]:SWEep:TIME <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**[:SENSe]:SWEep:TIME?**

Title: Sweep Time

Description: Sets the value of the minimum sweep time parameter. The sweep will complete in the shortest time possible greater than the specified time. To sweep as fast as possible, enter the minimum value allowed for the sweep time. Applies to zero span sweeps only.

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 30000 ms

Default Unit: ms

Range: 0.02 ms to 60000 ms

**[:SENSe]:SYNCtype <SS>**
**[:SENSe]:SYNCtype?**

Title: Sync Type

Description: Sets or queries the sync type, currently only SS is available for 5GNR

Parameter(s): <SS>

Query Return: SS

Default Value: SS

**[:SENSe]:TDOMain:DISPlay:LENGth <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**[:SENSe]:TDOMain:DISPlay:LENGth?**

Title: Time Domain Display Length

Description: Sets or queries the duration of the Time Domain measurements display's measurement. Available values are limited to 100us - 200ms.

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 20 ms

Default Unit: ms

Range: 0.1 ms to 200 ms

**[:SENSe]:TDOMain:DISPlay:STARt <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**[:SENSe]:TDOMain:DISPlay:STARt?**

Title: Time Domain Display Start Time

Description: Sets or queries the start time of the display for the Time Domain measurement. Available values are limited to 0 - 199.9ms.

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 0 ms

Default Unit: ms

Range: 0 ms to 199.9 ms

**[:SENSe]:TDOMain:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:TDOMain:STATe?**

Title: Time Domain State

Description: Enables and disables time domain mode

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:TDOMain:TRACe[:DATa]? [<ASCii|INTeger|REAL>],[<numeric_value>]**

Title: Time Domain Trace

Description: This command transfers the time domain trace data from the unit to the controller. The data format and optional parameters are identical to the TRACE[:DTATA]:ALL and the :FETCh[:DATA]:ALL command. See the document on those commands for more info.

Parameter(s): [<ASCii|INTeger|REAL>],[<numeric_value>]

**[:SENSe]:TDOMain:TRACe:DETector[:FUNCtion]**
**<POSitive|RMS|NEGative|SAMPle|NORMal>**
**[:SENSe]:TDOMain:TRACe:DETector[:FUNCtion]?**

Title: TDomain Detector Type

Description: Query and Set the time domain detector type. The detection type determines how the display point is derived from its associated measurements. POSitive Peak detection displays the maximum value of the associated measurements. RMS detection displays the average power of the associated measurements. NEGative Peak detection displays the minimum value of the associated measurements. NORMal detection displays the maximum value, POSitive Peak, when the point is odd and minimum value, NEGative Peak, when the point is even. SAMPle detection selects one of the associated measurement points to display.

Parameter(s): <POSitive | RMS | NEGative | SAMPle | NORMal>

Query Return: POS | RMS | NEG | SAMP | NORM

Default Value: POSitive

**[:SENSe]:TDOMain:TRACe:DISPlay[:STATe] <0 | 1 | ON | OFF>**
**[:SENSe]:TDOMain:TRACe:DISPlay[:STATe]?**

Title: TDomain Display State

Description: The trace visibility state. If it is OFF, the [:SENSe]:TDOMain:TRACe[:DATa]? command will return nan.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:TDOMain:TRACe:TYPE**
**<NORMal|MINimum|MAXimum|AVERage|RMAXimum|RMINimum|RAVerage|MATH>**
**[:SENSe]:TDOMain:TRACe:TYPE?**

Title: TDOMain Trace Type

Description: Specifies how successive sweeps are combined to produce the resulting display value. Setting the TYPE to NORMal will cause the displayed value for a point to be the current measured value for that point. Setting the TYPE to AVERage will cause the displayed value for a point to be the average of the last <integer> measured values where <integer> is set by [:SENSe]:AVERage:COUNt. Setting the TYPE to MAXimum will cause the displayed value for a point is the maximum measured value for that point over sweeps. Setting the TYPE to MINimum will cause the displayed value for a point is the minimum measured value for that point over sweeps.Setting the TYPE to RMAXimum will cause the displayed value for a point to be the maximum of the last <integer> measured values where <integer> is set by [:SENSe]:AVERage:COUNt.Setting the TYPE to RMINimum will cause the displayed value for a point to be the minimum of the last <integer> measured values where <integer> is set by [:SENSe]:AVERage:COUNt.

Parameter(s):

<NORMal | MINimum | MAXimum | AVERage | RMAXimum | RMINimum | RAVerage | MATH>

Query Return: NORM | MIN | MAX | AVER | RMAX | RMIN | RAV | MATH

Default Value: NORMal

**[:SENSe]:TDOMain:TRACe:UPDate[:STATe] <0 | 1 | ON | OFF>**
**[:SENSe]:TDOMain:TRACe:UPDate[:STATe]?**

Title: TDomain Update State

Description: The trace update state determine whether the time domain trace is update in every sweep.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:THD:FREQuency:FUNDamental <numeric_value> {HZ | KHZ | MHZ | GHZ}**
**[:SENSe]:THD:FREQuency:FUNDamental?**

Title: THD Fundamental Frequency

Description: Sets and queries the THD Fundamental Frequency for use in the measurement. When the fundamental auto is enabled then this value is not used, see setting THD:FREQ:FUND:AUTO.

Parameter(s): <numeric_value> {HZ | KHZ | MHZ | GHZ}

Query Return: Numeric (Hz)

Default Value: 3000000000 Hz

Default Unit: Hz

Range: 1 Hz to 400000000000 Hz

**[:SENSe]:THD:FREQuency:FUNDamental:AUTO <0 | 1 | ON | OFF>**
**[:SENSe]:THD:FREQuency:FUNDamental:AUTO?**

Title: THD Fundamental Frequency Auto

Description: This command can switches between automatic or manual selection of the THD fundamental frequency used. When automatic selection then the peak in the spectrum is used as the fundamental frequency. When switching the auto state off, the current modulation frequency result is used see :THD:FREQ:FUND

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:THD:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:THD:STATe?**

Title: THD Measurement State

Description: Sets and queries the state of THD Measurement

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**[:SENSe]:UPDown:CONFig <numeric_value>**
**[:SENSe]:UPDown:CONFig?**

Title: Up/Down Link Config

Description: Sets or queries the current up/down link config setting. This setting is only available when duplex type is set to TDD

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 0

Range: 0 to 6

**[:SENSe]:WCDMa:DEMod:CPICh <numeric_value> {<amplitude_units>}**
**[:SENSe]:WCDMa:DEMod:CPICh?**

Title: CPICH Power

Description: Set or query the power of the Common Pilot Channel for W-CDMA code domain demodulation measurements.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {<amplitude_units>}

Query Return: Numeric (<amplitude_units>)

Default Value: 30 dBm

Default Unit: dBm

Range: -300 dBm to 100 dBm

**[:SENSe]:WCDMa:DEMod:CPICh:STATe <0 | 1 | ON | OFF>**
**[:SENSe]:WCDMa:DEMod:CPICh:STATe?**

Title: CPICH Sync State

Description: Set or query whether the CPICH is used for synchronization during W-CDMA code domain demodulation measurements. Setting the value to ON or 1 enables use of the CPICH for synchronization. Setting the value to OFF or 0 disables use of the CPICH for synchronization.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:WCDMa:DEMod:PICH:CODE <numeric_value>**
**[:SENSe]:WCDMa:DEMod:PICH:CODE?**

Title: PICH Code Number

Description: Set or query the channelization code number of the PICH channel for W-CDMA code domain demodulation measurements.

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 16

Range: 0 to 255

**[:SENSe]:WCDMa:DEMod:SCCPch:CODE <numeric_value>**
**[:SENSe]:WCDMa:DEMod:SCCPch:CODE?**

Title: S-CCPCH Code Number

Description: Set or query the channelization code number of the S-CCPCH channel for W-CDMA code domain demodulation measurements.

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 3

Range: 1 to 255

**[:SENSe]:WCDMa:DEMod:SCCPch:SFACtor <string>**
**[:SENSe]:WCDMa:DEMod:SCCPch:SFACtor?**

Title: S-CCPCH Spreading Factor

Description: Set or query the spreading factor of the S-CCPCH channel for W-CDMA code domain demodulation measurements.

Parameter(s): <string>

Default Value: 256

**[:SENSe]:WCDMa:DEMod:SCODe <numeric_value>**
**[:SENSe]:WCDMa:DEMod:SCODe?**

Title: Scrambling Code

Description: Set or query the primary scrambling code used for W-CDMA code domain demodulation measurements. When the auto scrambling code feature is turned on, see [:SENSe]:WCDMa:DEMod:SCODe:AUTO, this setting is overwritten by the automatically detected value.

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 0

Range: -1 to 511

**[:SENSe]:WCDMa:DEMod:SCODe:AUTO <0 | 1 | ON | OFF>**
**[:SENSe]:WCDMa:DEMod:SCODe:AUTO?**

Title: Automatic Scrambling Code Detection

Description: Set or query the state of the automatic scrambling code detection feature. Setting the value to ON or 1 results in the instrument detecting the strongest signal present and using the scrambling code of that signal for demodulation. Setting the value to OFF or 0 applies the user-selected scrambling code value, see [:SENSe]:WCDMa:DEMod:SCODe.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:WCDMa:DEMod:SFACtor <string>**
**[:SENSe]:WCDMa:DEMod:SFACtor?**

Title: Maximum Spreading Factor

Description: Set or query the maximum spreading factor used when demodulating a W-CDMA signal.

Parameter(s): <string>

Default Value: 512

**[:SENSe]:WCDMa:DEMod:THRShold <numeric_value> {DB}**
**[:SENSe]:WCDMa:DEMod:THRShold?**

Title: Active Code Threshold

Description: Set or query the threshold used for detecting active traffic channels in W-CDMA code domain demodulation measurements. If the automatic threshold feature is enabled, see [:SENSe]:DEMod:THRShold:AUTO, this setting is overwritten by the automatically detected threshold.

Parameter(s): <numeric_value> {DB}

Query Return: Numeric (dB)

Default Value: -30 dB

Default Unit: dB

Range: -50 dB to -10 dB

**[:SENSe]:WCDMa:DEMod:THRShold:AUTO <0 | 1 | ON | OFF>**
**[:SENSe]:WCDMa:DEMod:THRShold:AUTO?**

Title: Automatic Threshold

Description: Set or query the state of the Automatic Threshold feature when demodulating a W-CDMA signal. Setting the value to ON or 1 enables the automatic threshold feature. Setting the value to OFF or 0 disables the automatic threshold feature and applies the user-selected threshold, see [:SENSe]:WCDMa:DEMod:THRShold.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: ON

**[:SENSe]:WCDMa:DEMod:TXPower <numeric_value> {<amplitude_units>}**
**[:SENSe]:WCDMa:DEMod:TXPower?**

Title: Maximum Transmit Power

Description: Set or query the maximum transmit power of the base station for W-CDMA code domain demodulation measurements.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {<amplitude_units>}

Query Return: Numeric (<amplitude_units>)

Default Value: 43 dBm

Default Unit: dBm

Range: -300 dBm to 100 dBm

**[:SENSe]:WCDMa:SYNC?**

Title: Get Sync Measurements

Description: Returns the last stored sync measurement results.

This command returns frequency error stats, carrier id, corrected carrier frequency, frame start, frequency error, gps info, scrambling code, hardware status and time offset(in ns) as output in json format.

This command does not wait for the current measurement to complete before returning results.

An example is as follows:

{
"FrequencyErrorStats": {
"Count": 185364,
"Deviation": 1.8153972412361203,
"DeviationUnits": "Hz",
"Max": 117.21477065686643,
"Mean": 110.26521356758234,
"Min": 107.69124579804725,
"Units": "Hz"
},
"anritsuHardwareStatus": {
"hardwareErrors": {
"ADCOverRange": 0
},
"statusMessage": "statusSyncDone",
"statusNumber": 10
},
"carrierId": 1,
"correctedCarrierFrequency": 1000000111.0165963,
"frameStart": 39828,
"frequencyError": 111.01659632196692,
"gps": {
"coordinates": "37.146585, -121.656618,128m",
"time": "Tue Jan 1 00:00:00 2024 GMT"
},
"pccpchAbsPower": -48.427171521027155,
"pccpchDeltaPower": -0.3524818420410156,
"pccpchRelPower": -11.313854217529297,
"pcpichEvm": 0.7005375027656555,
"pcpichPower": -48.074689327855744,
"pcpichRelativePower": -10.961372375488281,
"pichAbsPower": -56.04133975526693,

"pichDeltaPower": -7.966650009155273,

"pichRelPower": -18.928022384643555,

"ppmFrequencyError": 0.1110165963219669,

"sccpchAbsPower": -55.5802791084507,

"sccpchDeltaPower": -7.505590438842773,

"sccpchRelPower": -18.466962814331055,

"scramblingCode": 0,

"scramblingCodeGroup": 0,

"timeOffset": 3753108.716643614,

"units": "DBM"

}

### :STATus:OPERation[:EVENt]?

Title: Get Operation Status

Description: This command requests information about the current status of the instrument. Each bit of the return value represents some operation. Only a subset of the bits are implemented for each application. The number returned is the decimal representation of the bit-wise OR of the enabled bits:

Bit Decimal Value_Description

0 1 Not implemented

1 2 Not implemented

2 4 Not implemented

3 8 Not implemented

4 16 Not implemented

5 32 Not implemented

6 64 Not implemented

7 128 Not implemented

8 256 Sweep Complete

This bit is set to 0 when the command :INITiate[:IMMediate] is sent to trigger a sweep. It will have a value of 1 when the sweep has completed.

9 512 I/Q Capture

This bit indicates whether the instrument is currently capturing I/Q data. It is set to 1 when the MEAS:IQ:CAPT command is issued. This bit will be set to 0 when the capture is completed normally (in block mode), or is aborted, either due to the ABORt command or some other command which invalidates the capture.

10 1024 Not implemented

11 2048 Not implemented

12 4096 Not implemented

13 8192 Not implemented

14 16384 Not implemented

15 0 Will always be 0

**:THReshold:SINR <numeric_value> {DB}**
**:THReshold:SINR?**

Title: Sinr threshold

Description: Sets or queries SINR threshold for display. Values below the threshold will not be shown. This only affects the display and does not change the contents of any other measurement data SCPI queries.

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {DB}

Query Return: Numeric (dB)

Default Value: 6 dB

Default Unit: dB

Range: -50 dB to 50 dB

**:TRACe<n>:IQ:DATA? [<numeric_value>]**

Title: IQ Data

Description: This command transfers IQ data from the instrument to the controller. Data is transferred from the instrument as an IEEE definite length arbitrary block response, which has the form <header><block>.

For a detailed description of the response format, see TRACe:IQ:DATA:FORMat?.

If IQ:MODE? is STREAM, this command will block until the next block of streaming data is available, then return it.

If IQ:MODE? is SINGle, this command will return data immediately if a capture has been already completed, or it will wait for an in-progress capture to complete before returning data, or it will return #0 if a capture has never been started.

Regardless of capture mode, this command will return #0 if an error condition is encountered during an in-progress capture (see MEAS:IQ:CAPT). Clients should check the SCPI error queue with SYST:ERR:NEXT? to determine what action to take.

In streaming mode there is an optional numeric parameter for the max chunks of data to return for cases where the read can't keep up.

Note that once data is requested, all of it must be read on the controller side. The instrument will block further SCPI processing until each data transfer is complete. If a data transfer remains incomplete for long enough it could cause error messages to be displayed until either all of the data is read or the connection where the transfer is taking place is closed.

Parameter(s): [<numeric_value>]

TRACe Suffix Range: 0, Default = 0

**:TRACe<n>:DETector[:FUNCtion] <POSitive|RMS|NEGative|SAMPle|NORMal>**
**:TRACe<n>:DETector[:FUNCtion]?**

Title: Trace Detector Type

Description: Sets the detection method for the specific trace. The detection type determines how the display point is derived from its associated measurements. POSitive Peak detection displays the maximum value of the associated measurements. RMS detection displays the average power of the associated measurements. NEGative Peak detection displays the minimum value of the associated measurements. SAMPle detection selects one of the associated measurement points to display. NORMal detection displays the maximum value, POSitive Peak, when the point is odd and minimum value, NEGative Peak, when the point is even. This command will be ignored when spectrogram is enabled by DISPlay:VIEW. In zero span and RTSA mode the SAMPle detection replaces RMS detection.

Parameter(s): <POSitive | RMS | NEGative | SAMPle | NORMal>

Query Return: POS | RMS | NEG | SAMP | NORM

TRACe Suffix Range: 1-6, Default = 1

Default Value: POSitive

**:TRACe<n>:DISPlay[:STATe] <0 | 1 | ON | OFF>**
**:TRACe<n>:DISPlay[:STATe]?**

Title: Trace Display State

Description: The trace visibility state. If it is OFF, the :TRAC:DATA? command will return nan.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

TRACe Suffix Range: 1-6, Default = 1

Default Value: ON

**:TRACe<n>:DISPlay:VIEW:SPECtrogram:ELAPsed?**

Title: Spectrogram Elapsed

Description: Provides the index and time values for the specified trace based on which trace from the spectrogram history it represents.

TRACe Suffix Range: 1-6, Default = 1

**:TRACe<n>:DISPlay:VIEW:SPECtrogram:POSition <numeric_value>**
**:TRACe<n>:DISPlay:VIEW:SPECtrogram:POSition?**

Title: Spectrogram Trace Position

Description: The command sets the spectrogram history index that specific trace referencing to.

Parameter(s): <numeric_value>

Query Return: Numeric

TRACe Suffix Range: 1-6, Default = 1

Default Value: 0

Range: 0 to 600

**:TRACe<n>:DISPlay:VIEW:SPECtrogram:TIME <numeric_value>**
**:TRACe<n>:DISPlay:VIEW:SPECtrogram:TIME?**

Title: Spectrogram Trace Time

Description: The command select the spectrogram history trace base on elapsed time from the newest trace.

Parameter(s): <numeric_value>

Query Return: Numeric (ms)

TRACe Suffix Range: 1-6, Default = 1

Default Value: 0 ms

Default Unit: ms

Range: 0 ms to 100000000 ms

**:TRACe<n>:NORMalize <0 | 1 | ON | OFF>**
**:TRACe<n>:NORMalize?**

Title: Trace Normalization State

Description: Turns normalization on or off for the specified trace. When normalization is turned on, the last completed sweep will be stored and subtracted from the live data until normalization is turned off. This will cancel out whatever was on the trace at that time and result in the trace flattening and moving to 0 dBm.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

TRACe Suffix Range: 1-6, Default = 1

Default Value: OFF

**:TRACe<n>:SWEep:COUNt[:CURRent]?**

Title: Trace Sweep Count

Description: The trace sweep count returns current sweep count which can be useful for multiple-sweep measurements like average and min/max hold.

Query Return: Numeric

TRACe Suffix Range: 1-6, Default = 1

Default Value: 0

Range: 0 to 4000000000

**:TRACe<n>:TYPE**
**<NORMal|MINimum|MAXimum|AVERage|RMAXimum|RMINimum|RAVerage|MATH>**
**:TRACe<n>:TYPE?**

Title: Trace Type

Description: Specifies how successive sweeps are combined to produce the resulting display value. Setting the TYPE to NORMal will cause the displayed value for a point to be the current measured value for that point. Setting the TYPE to AVERage will cause the displayed value for a point to be the average of the last <integer> measured values where <integer> is set by [:SENSe]:AVERage:COUNt. Setting the TYPE to MAXimum will cause the displayed value for a point is the maximum measured value for that point over sweeps. Setting the TYPE to MINimum will cause the displayed value for a point is the minimum measured value for that point over sweeps.Setting the TYPE to RMAXimum will cause the displayed value for a point to be the maximum of the last <integer> measured values where <integer> is set by [:SENSe]:AVERage:COUNt.Setting the TYPE to RMINimum will cause the displayed value for a point to be the minimum of the last <integer> measured values where <integer> is set by [:SENSe]:AVERage:COUNt. Setting the type to MATH only applies to traces 5 and 6 The math operation performed is always a division of the linear powers (subtraction of the log values) of the traces no matter what units selected. For trace 5 the operation is trace1/trace2 (trace1 -trace2 in log units) For trace 6 the operation is trace2/trace1 (trace2 - trace1 in log units) This command will be ignored when spectrogram is enabled by DISPlay:VIEW.

Parameter(s):

<NORMal | MINimum | MAXimum | AVERage | RMAXimum | RMINimum | RAVerage | MATH>

Query Return: NORM | MIN | MAX | AVER | RMAX | RMIN | RAV | MATH

TRACe Suffix Range: 1-6, Default = 1

Default Value: NORMal

**:TRACe<n>:UPDate[:STATe] <0 | 1 | ON | OFF>**
**:TRACe<n>:UPDate[:STATe]?**

Title: Trace Update State

Description: The trace update state detemine whether the trace is update in every sweep.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

TRACe Suffix Range: 1-6, Default = 1

Default Value: ON

**:TRACe:CLEar <numeric_value>**

Title: Trace Clear

Description: This command clears the trace's history and current sweep data. Trace history refers to the previous trace data used to calculate trace types such as min/max hold and averaging.

Parameter(s): <numeric_value>

**:TRACe:CLEar:ALL**

Title: All Traces Clear

Description: This command clears the trace history and current sweep data for all traces. Trace history refers to the previous data used to calculate trace types such as min/max hold and averaging.

**:TRACe[:DATA]? <numeric_value>**

Title: Trace Data

Description: This command transfers trace data from the instrument to the controller. Data is transferred from the instrument as an IEEE definite length arbitrary block response, which has the form <header><block>.

This command takes a single integer parameter specifying the trace number to transfer. If the parameter value is out of the range of valid trace numbers, the first trace (1) will be transferred. The binary block header specifies the number of data bytes. It looks like #AX, where A is the number of digits in X and X is the number of bytes in the <block>.

The first character is the literal ascii hash '#' 043.

The second character (A) is a single ascii digit '1' to '9' describing the number of bytes in the length section (X). This number is called nlength.

The next nlength bytes make up an ascii string of digits '1' to '9' describing the length of the <block> data.

For example, if the first 6 bytes are #49999, then the nlength is 4. The 4 bytes of length are 9999. After that follows the <block>, which would be 9999 bytes in size.

The format of the block data depends on the current data format setting (FORMat[:TRACe][:DATA]). The number of amplitudes returned is equal to the current number of display points (DISPlay:POINtcount).

The query command will return a #0 if data is invalid for the active trace.

Parameter(s): <numeric_value>

**:TRACe[:DATA]:ALL? [<ASCii|INTeger|REAL>],[<numeric_value>]**

Title: Trace Data All

Description: This command transfers trace data from the instrument to the controller for all traces with display state ON (see Trace Display State commands).

The traces are returned back to back with no delimiter between traces using the the same IEEE definite length arbitrary block response of the Trace Data command.

However, after the usual header this command includes additional header information to support parsing the following data. This additional header consists of a two byte integer indicating the number of display points per trace as well as a two byte trace flags value.

The bits in the two byte trace flags value are mapped to a trace number and indicate whether that trace is included in the following data. Bit 0 is the least significant bit and is unused. Bits 1 through N are mapped to traces 1 through N. If the bit value is 1, the trace data for that trace is included in the following data. If the bit value is 0, the trace data for that trace is not included in the following data.

For example, a trace flags value of 0x4C indicates that trace data for traces 2, 3, and 6 are included in the following data. The traces are returned in increasing order. The format of the data follows the Trace Data Format setting.

An optional parameter can be given to set the format data output, the parameter follows the same enumerable value as the Trace Data format setting. When using REAL as the output format, the numeric value field is required. See the document on Trace Data Format (:FORMat[:TRACe][:DATA]) for more info.

Parameter(s): [<ASCii|INTeger|REAL>],[<numeric_value>]

**:TRACe:IQ:DATA:FORMat <PACKed>**
**:TRACe:IQ:DATA:FORMat?**

Title: IQ Data Format

Description: This command selects the data format for transferring I/Q data via the TRACe:IQ:DATA? query.

Supported data formats include PACKed, which is a binary format that includes embedded timestamps, and ASCii, which is a human-readable, comma-delimited list of samples. The PACKed format is recommended for applications that require precision timestamps, high data throughput, and processing in real time, such as TDOA. The ASCii format can be much slower to transfer, but it has the advantage of being human readable.

When the data format is PACKed, TRAC:IQ:DATA? query uses SCPI standard (IEEE 488.2) definite length block data format for responses. The data format is '#AXD', where X is one or more ASCII digits specifying the number of bytes in D, and A is a single ASCII digit specifying the number of digits in X. D contains binary data. The whole 'D' part looks like 'L\nB', where L is an ASCII string of the form 'latitude, longitude' in decimal degrees, '\n' is a single byte newline delimiter marking the end of the GPS location component, and B is the I/Q data taken from the instrument's RAM. The binary structure of B includes timestamps embedded within the samples and is described in detail in the user guide of the instrument. For MS2090A, the format information can be found in chapter 7, section 3 (7-3 IQ Data Format) of the user guide. For MS2710XA, the format information can be found in Chapter 5, section 9 (5-9 I/Q Capture Block Mode) of the user guide. For MS2720XA, the format information can be found in chapter 6, section 3 (6-3 IQ Data Format) of the user guide. The values of the GPS latitude and longitude in the header are undefined if the GPS is not actually fixed (i.e. FETCH:GPS? returns "NO FIX").

<MS2710xA>

When the data format is ASCii, TRAC:IQ:DATA? query returns an definite length block data response in Comma Separated Values (CSV) Standard File Format, which is easily imported into popular spreadsheet programs:

* Each record is on one line

* Lines are separated by carriage return and line feed (CRLF)

* Fields are separated by commas

* Trailing and leading whitespace is insignificant

* No quotes, embedded commas, or embedded newlines in this output

The data format is '#AXD' where X is one or more ASCII digits specifying the number of bytes in D, and A is the number of digits in X. D contains a list of I/Q samples. Each of I and Q are formatted as signed decimal integers. Each I is separated from its corresponding Q by a comma. Adjacent samples are separated by newlines. So, the data (excluding the '#AX' header) looks like:

I1,Q1

I2,Q2

I3,Q3

...

The ASCii format response contains neither time stamp nor header. To get the GPS location, see the FETCh:GPS? query.

Parameter(s): <PACKed>

Query Return: PACK

Default Value: PACKed

**:TRACe:PLAYback <PLAY|STOP|PAUSe>**
**:TRACe:PLAYback?**

Title: Trace Playback State

Description: This command sets or queries the Trace Playback state, which can be STARt, STOP, or PAUSe.

Parameter(s): <PLAY|STOP|PAUSe>

Query Return: PLAY|STOP|PAUS

Default Value: STOP

**:TRACe:PLAYback:FILE?**

Title: Current Playback File

Description: This command queries the currently loaded Trace Recording file's name used for trace playback, e.g. "RecordedSession_20210901T143210.786.fmrec"

Default Value: None

**:TRACe:PLAYback:INTerval <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**:TRACe:PLAYback:INTerval?**

Title: Trace Playback Interval

Description: This command sets or queries the playback interval rate at which trace recording are being played.

The interval rate can be set from 20 milliseconds up to 5 seconds.

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 1000 ms

Default Unit: ms

Range: 20 ms to 5000 ms

**:TRACe:PLAYback:LOOP <0 | 1 | ON | OFF>**
**:TRACe:PLAYback:LOOP?**

Title: Loop Playback

Description: This command sets or queries whether playback will restart at the beginning when it reaches the end of the playback file.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:TRACe:PLAYback:POSition?**

Title: Playback Position

Description: This command supplies information related to the current position of a playback session. It returns the total number of traces in the session in addition to the index of the last played trace along with the timestamp value as the number of milliseconds since 1970-01-01T00:00:00.000 Coordinated Universal Time(UTC) describing when the trace was recorded. The geographical position properties hold the latitude and longitude in decimal degrees. A response might look like the following:

{
"traceCount":1500,
"trace":
{
"index":123,
"timestamp":1605311779428,
"latitude":37.146880,
"longitude":-121.656113
}
}

If there isn't an active playback session all values will be 0.

Default Value: {"index":0,"timestamp":0}

**:TRACe:PLAYback:SEEK <numeric_value>**
**:TRACe:PLAYback:SEEK?**

Title: Trace Playback Seek

Description: This command sets and queries the current position in a playback session. A playback session contains a list of recorded traces and the zero-based index used by this command represents the trace at that index in the list, where the trace at index 0 is the trace that was recorded first and the trace at the last index is the trace recorded last.

The set command takes an index which will be the index of the next played trace. The query command takes no parameters and returns the index of the last played trace.

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 0

Range: 0 to 0

**:TRACe:PLAYback:TAG <NEXT|PREVious|ADD|REMove|ALL>**

Title: Trace Playback Tag

Description: This command skips to the next or previous tag in the playback, if one exists. It can also be used to add/remove tags in the current playback session, which will modify the session file so that the tags persist the next time it's played. The ALL option allows for only the tag traces to be played in the recording. The tag traces are only played in ascending order in which they were created.

Parameter(s): <NEXT | PREVious | ADD | REMove | ALL>

**:TRACe:PLAYback:TAGS? [<RECord|PLAYback>]**

Title: Tag Info

Description: This command returns which traces are tagged in a record or playback session. The response is in JSON format. Each tag is represented as an index, timestamp and geographical position. The indices correspond to the position of the tagged trace in the full recorded history, the timestamp values are millseconds since 1970-01-01T00:00:00.000 Coordinated Universal Time(UTC) describing when the trace was recorded, and the geographical position properties hold the latitude and longitude in decimal degrees. The index values can be used as a parameter to the TRACe:PLAYback:SEEK command. A response might look like the following:

{

"tags":

[

{"index":3,"timestamp":1605310939565,"latitude":37.146880,"longitude":-121.656113},

{"index":75,"timestamp":1605311015834,"latitude":37.146880,"longitude":-121.656113}

]

}

If there is no playback or record session active, the response will be {"tags":[]} An optional parameter can be set to query the record or playback tag list.

Parameter(s): [<RECord | PLAYback>]

Default Value: {"tags":[]}

**:TRACe:PLAYback:TAG:TABLe <0 | 1 | ON | OFF>**
**:TRACe:PLAYback:TAG:TABLe?**

Title: Playback Table

Description: Frontend command to indicate tag table on/off

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:TRACe:PRESet:ALL**

Title: Preset All Trace

Description: This command preset all traces which turn Traces 2-6 off and set Trace 1 to Clear/Write, Active, Peak Detector.

**:TRACe:RECord <RECord|STOP|PAUSe>**
**:TRACe:RECord?**

Title: Trace Recording State

Description: This command sets or queries the Trace Recording state, which can be RECord, STOP, or PAUSe.

Parameter(s): <RECord | STOP | PAUSe>

Query Return: REC | STOP | PAUS

Default Value: STOP

**:TRACe:RECord:CSV:LINK <RECord|PLAYback|OFF>**
**:TRACe:RECord:CSV:LINK?**

Title: Enable CSV Recording during Trace Record or Trace Playback

Description: This command sets or queries whether Trace Recording in CSV is linked to the Trace Record or Trace Playback states,

which can be RECord, PLAYback or OFF.

RECord:

Recording CSV State (TRACe:RECord:CSV:STATe) will be updated base on Trace Record State (TRACe:RECord) changes as following

Record State: Record CSV State:

RECord RECord

PAUSe PAUSe

STOP STOP

PLAYback:

Recording CSV State will be updated base on on Trace Playback State (TRACe:PLAYback) changes as following

Playback State: Record CSV State:

PLAY RECord

PAUSe PAUSe

STOP STOP

OFF:

Recording CSV State remain unaffected by Trace Record State or Trace Playback State changes.

Parameter(s): <RECord|PLAYback|OFF>

Query Return: REC|PLAY|OFF

Default Value: OFF

**:TRACe:RECord:CSV:STATe <RECord|STOP|PAUSe>**
**:TRACe:RECord:CSV:STATe?**

Title: Trace Recording CSV State

Description: This command sets or queries whether Trace Recording will also write a CSV of the recorded data, which can be RECord, STOP, or PAUSe.

Parameter(s): <RECord|STOP|PAUSe>

Query Return: REC|STOP|PAUS

Default Value: STOP

**:TRACe:RECord:INTerval:MODE <AUTO|MANual>**
**:TRACe:RECord:INTerval:MODE?**

Title: Trace Recording Interval Mode

Description: This command sets or queries whether Trace Recording will AUTomatically record each trace or use MANual intervals.

Parameter(s): <AUTO|MANual>

Query Return: AUTO|MAN

Default Value: AUTO

**:TRACe:RECord:INTerval:TIME <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**:TRACe:RECord:INTerval:TIME?**

Title: Trace Recording Interval Time

Description: This command sets or queries the interval length for use with manual interval mode Trace Recording.

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 10000 ms

Default Unit: ms

Range: 20 ms to 3600000 ms

**:TRACe:RECord:LINK <0 | 1 | ON | OFF>**
**:TRACe:RECord:LINK?**

Title: Enable Recording Link Playback

Description: This command sets or queries whether a Trace Recording is linked to Trace Playback states. Recording State will be updated base on on Trace Playback State (TRACe:PLAYback) changes as following

Playback State: Record State:

PLAY RECord

PAUSe STOP

STOP STOP

For example, this can be used to capture tag traces (TRACe:PLAYback:TAG) or normal traces from one file to a new recording file.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:TRACe:RECord:MODE <AUTO|TIMed>**
**:TRACe:RECord:MODE?**

Title: Trace Recording Mode

Description: This command sets or queries the Trace Recording mode, AUTO or TIMed. AUTO mode means recording will continue indefinitely. TIMed mode means recording will last for the specified recording time set by the Trace Recording Time setting.

Parameter(s): <AUTO|TIMed>

Query Return: AUTO|TIM

Default Value: AUTO

**:TRACe:RECord:TAG**

Title: Trace Recording Add Tag

Description: This command adds a tag to the current trace as it is being recorded or played.

**:TRACe:RECord:TIME <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**:TRACe:RECord:TIME?**

Title: Trace Recording Time

Description: This command sets or queries the amount of time to record for if the Trace Recording mode is TIMed.

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 20000 ms

Default Unit: ms

Range: 1000 ms to 1000000000 ms

**:TRACe:SELect <numeric_value>**
**:TRACe:SELect?**

Title: Select Trace

Description: The selected trace will be used by operations that use a single trace. The max number of traces available to select is model specific.

To display the trace, see TRACe<n>:DISPlay[:STATe]

To update the trace state, see TRACe<n>:UPDate[:STATe]

Parameter(s): <numeric_value>

Query Return: Numeric

Default Value: 1

Range: 1 to 6

**:TRACe:SPECtrogram:DETector[:FUNCtion]**
**<POSitive|RMS|NEGative|SAMPle|NORMal>**
**:TRACe:SPECtrogram:DETector[:FUNCtion]?**

Title: Spectrogram Detector Type

Description: Sets the detection method for the spectrogram trace. The detection type determines how the display point is derived from its associated measurements. POSitive Peak detection displays the maximum value of the associated measurements. RMS detection displays the average power of the associated measurements. NEGative Peak detection displays the minimum value of the associated measurements. NORMal detection displays the maximum value, POSitive Peak, when the point is odd and minimum value, NEGative Peak, when the point is even. SAMPle detection selects one of the associated measurement points to display. This command will be ignored when spectrogram is disabled by DISPlay:VIEW. In RTSA mode the SAMPle detection replaces RMS detection.

Parameter(s): <POSitive | RMS | NEGative | SAMPle | NORMal>

Query Return: POS | RMS | NEG | SAMP | NORM

Default Value: POSitive

**:TRACe:SPECtrogram:TYPE**
**<NORMal|MINimum|MAXimum|AVERage|RMAXimum|RMINimum|RAVerage>**
**:TRACe:SPECtrogram:TYPE?**

Title: Spectrogram Trace Type

Description: Specifies how successive sweeps are combined to produce the resulting display value. Setting the TYPE to NORMal will cause the displayed value for a point to be the current measured value for that point. Setting the TYPE to AVERage will cause the displayed value for a point to be the average of the last <integer> measured values where <integer> is set by [:SENSe]:AVERage:COUNt. Setting the TYPE to MAXimum will cause the displayed value for a point is the maximum measured value for that point over sweeps. Setting the TYPE to MINimum will cause the displayed value for a point is the minimum measured value for that point over sweeps.Setting the TYPE to RMAXimum will cause the displayed value for a point to be the maximum of the last <integer> measured values where <integer> is set by [:SENSe]:AVERage:COUNt.Setting the TYPE to RMINimum will cause the displayed value for a point to be the minimum of the last <integer> measured values where <integer> is set by [:SENSe]:AVERage:COUNt. This command will be ignored when spectrogram is disabled by DISPlay:VIEW.

Parameter(s): <NORMal | MINimum | MAXimum | AVERage | RMAXimum | RMINimum | RAVerage>

Query Return: NORM | MIN | MAX | AVER | RMAX | RMIN | RAV

Default Value: NORMal

**:TRACe:STATus? <numeric_value>**

Title: Trace Status

Description: This command returns a response of the same format as a valid TRACe[:DATA] response, except that instead of amplitude, each comma-delimited value is a decimal integer representing the bitwise-OR of one or more status bits. Each bit of the integer is set according to the table below to indicate that the corresponding trace point has the indicated status:

OkStatus = 0x0

ADCOverrange = 0x1

Saturation = 0x2

SLOLockFailure = 0x4

LO1LockFailure = 0x8

LO2LockFailure = 0x10

TGLOLockFailure = 0x20

LimitFailure = 0x40

OverpowerFailure = 0x1000

DeviceAccessFailure = 0x10000 Failed to communicate with the device.

EPROMFailure = 0x20000 Failed to access device EEPROM.

DataTransIrregular = 0x40000 Data transfer irregular.

NoCalibration = 0x80000 Data is not calibrated.

TGRefLockFailure = 0x100000 TG REF lock Failure

TGRfLockFailure = 0x200000 TG RF lock Failure

AlcUnlevelLow = 0x400000 ALC Unleveled Low

AlcUnlevelHigh = 0x800000 ALC Unleveled High

Parameter(s): <numeric_value>

**:TRIGger:GATE:SLOPe <POSitive|NEGative>**
**:TRIGger:GATE:SLOPe?**

Title: Gated Sweep Trigger Slope

Description: The gate trigger slope indicates whether the trigger point is on the rising or the falling edge of the external trigger signal.

Parameter(s): <POSitive|NEGative>

Query Return: POS|NEG

Default Value: POSitive

**:TRIGger[:SEQuence]:ATRigger <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**:TRIGger[:SEQuence]:ATRigger?**

Title: Trigger Auto Period

Description: The trigger auto period is the amount of time that must pass without an trigger event before a measurement is triggered automatically.

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 1000 ms

Default Unit: ms

Range: 0.001 ms to 5000 ms

**:TRIGger[:SEQuence]:ATRigger:STATe <0 | 1 | ON | OFF>**
**:TRIGger[:SEQuence]:ATRigger:STATe?**

Title: Trigger Auto State

Description: The trigger auto state indicates whether or not a measurement should be triggered automatically after the Trigger Auto Period elapses.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:TRIGger[:SEQuence]:DELay <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**:TRIGger[:SEQuence]:DELay?**

Title: Trigger Delay

Description: The trigger delay is the amount of time between when an external trigger event happens and when the measurement starts.

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 0 ms

Default Unit: ms

Range: -1000 ms to 1670 ms

**:TRIGger[:SEQuence]:DELay:STATe <0 | 1 | ON | OFF>**
**:TRIGger[:SEQuence]:DELay:STATe?**

Title: Trigger Delay State

Description: The trigger delay state turns trigger delay on/off.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:TRIGger[:SEQuence]:HOLDoff <numeric_value> {PS | NS | US | MS | S | MIN | HR}**
**:TRIGger[:SEQuence]:HOLDoff?**

Title: Trigger Holdoff

Description: The trigger holdoff is the amount of time between when a measurement ends and when to start looking for the next external trigger event.

Parameter(s): <numeric_value> {PS | NS | US | MS | S | MIN | HR}

Query Return: Numeric (ms)

Default Value: 0 ms

Default Unit: ms

Range: 0 ms to 5000 ms

**:TRIGger[:SEQuence]:HOLDoff:STATe <0 | 1 | ON | OFF>**
**:TRIGger[:SEQuence]:HOLDoff:STATe?**

Title: Trigger Holdoff State

Description: The trigger holdoff state turns trigger holdoff on/off.

Parameter(s): <0 | 1 | ON | OFF>

Query Return: 0 | 1

Default Value: OFF

**:TRIGger[:SEQuence]:SLOPe <POSitive|NEGative>**
**:TRIGger[:SEQuence]:SLOPe?**

Title: Trigger Slope

Description: The trigger slope indicates whether the trigger point is on the rising or the falling edge of the external trigger signal.

Parameter(s): <POSitive|NEGative>

Query Return: POS|NEG

Default Value: POSitive

**:TRIGger[:SEQuence]:SOURce <EXT1|EXT2|IMMediate|VIDeo>**
**:TRIGger[:SEQuence]:SOURce?**

Title: Trigger Source

Description: The trigger source indicates whether or not to look for a trigger condition before making measurements.

IMMediate trigger source means that sweeps will be triggered without looking for a trigger event (free-run). VIDeo trigger is only valid for zero span and IQ capture. Video trigger is not supported for IQ streaming. External trigger options with a suffix number in them are only available on instruments with more than one trigger port and in this case omitting the number will be the same as using EXT1. For products that support trigger port direction settings, the ports will only be available as trigger sources if the trigger direction for that port is set to IN.

Parameter(s): <EXT1|EXT2|IMMediate|VIDeo>

Query Return: EXT1|EXT2|IMM|VID

Default Value: IMMediate

**:TRIGger[:SEQuence]:VIDeo:HYSTeresis <numeric_value> {DB}**
**:TRIGger[:SEQuence]:VIDeo:HYSTeresis?**

Title: Trigger Video Hysteresis

Description: A relative amplitude value in dBm centered around the video trigger level which the input signal must cross for a trigger event to occur when trigger source is set to video (TRIG:SOUR VID).

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {DB}

Query Return: Numeric (dB)

Default Value: 0 dB

Default Unit: dB

Range: 0 dB to 200 dB

**:TRIGger[:SEQuence]:VIDeo:LEVel <numeric_value> {<amplitude_units>}**
**:TRIGger[:SEQuence]:VIDeo:LEVel?**

Title: Trigger Video Level

Description: The amplitude value which the input signal must cross for a trigger event to occur when trigger source is video (TRIG:SOUR VID).

See UNIT:POWer to query or set the current amplitude units.

Parameter(s): <numeric_value> {<amplitude_units>}

Query Return: Numeric (<amplitude_units>)

Default Value: 0 dBm

Default Unit: dBm

Range: -150 dBm to 30 dBm

**:UNIT:CHPower:PSDensity <HZ|MHZ>**
**:UNIT:CHPower:PSDensity?**

Title: Power Spectral Density Units

Description: Sets the power spectral density units to the current amplitude per either Hz or MHz. For example, if the current amplitude unit is dBm, then Hz would refer to dBm/Hz, and MHZ would be refer to dBm/MHz.

Parameters DBMHz and DBMMHz have been deprecated, but are still accepted values.

Parameter(s): <HZ|MHZ>

Query Return: HZ|MHZ

Default Value: HZ

**:UNIT:POWer**
**<DBM|DBM/M2|DBW|DBW/M2|DBV|DBV/M|DBMV|DBMV/M|DBUV|DBUV/M|DBA|DBA/M|V|V/M|A|A/M|W|W/M2|W/CM2>**
**:UNIT:POWer?**

Title: Measurement Units

Description: Sets the default amplitude units for input, output and display.

<MS2090A/MS2080A/MS2070A>

The following units are only available when Field Strength is ON: DBM/M2, DBW/M2, DBV/M, DBMV/M, DBUV/M, DBA/M, V/M, A/M, W/M2, W/CM2

To query or set the Field Strength state, use [:SENSe]:FSTRength:STATe

Parameter(s):

<DBM | DBM/M2 | DBW | DBW/M2 | DBV | DBV/M | DBMV | DBMV/M | DBUV | DBUV/M | DBA | DBA/M | V | V/M | A | A/M | W | W/M2 | W/CM2>

Query Return:

DBM | DBM/M2 | DBW | DBW/M2 | DBV | DBV/M | DBMV | DBMV/M | DBUV | DBUV/M | DBA | DBA/M | V | V/M | A | A/M | W | W/M2 | W/CM2

Default Value: DBM

**:UNIT:POWer**
**<DBM|DBM/M2|DBW|DBW/M2|DBV|DBV/M|DBMV|DBMV/M|DBUV|DBUV/M|DBA|DBA/M|V|V/M|A|A/M|W|W/M2|W/CM2>**
**:UNIT:POWer?**

Title: Measurement Units

Description: Sets the default amplitude units for input, output and display.

<MS2090A/MS2080A/MS2070A>

The following units are only available when Field Strength is ON: DBM/M2, DBW/M2, DBV/M, DBMV/M, DBUV/M, DBA/M, V/M, A/M, W/M2, W/CM2

To query or set the Field Strength state, use [:SENSe]:FSTRength:STATe

Parameter(s):

<DBM | DBM/M2 | DBW | DBW/M2 | DBV | DBV/M | DBMV | DBMV/M | DBUV | DBUV/M | DBA | DBA/M | V | V/M | A | A/M | W | W/M2 | W/CM2>

Query Return:

DBM | DBM/M2 | DBW | DBW/M2 | DBV | DBV/M | DBMV | DBMV/M | DBUV | DBUV/M | DBA | DBA/M | V | V/M | A | A/M | W | W/M2 | W/CM2

Default Value: DBM

2025-03-03T09:11:25PM

# Appendix A — SCPI Error Table

This section lists the error code/messages returned from the device when an error occurs during the execution of the SCPI command.

Error messages are classified by error number as listed in table below.

| Error Number | Error String | Description |
|---|---|---|
| 11 | EEPROM read/write error | EEPROM read and/or write failed |
| 10 | Hardware communication error | Failed to communicate with device hardware. |
| 2 | Option mismatch | Operation failed due to option mismatch. |
| 1 | Model mismatch | Operation failed due to option mismatch. |
| 0 | No error | The queue is completely empty. Every error/event in the queue has been read or the queue was purposely cleared by power-on, `*CLS`, etc. |
| -100 | Command error | Command error |
| -101 | Invalid character | A syntactic element contains a character which is invalid for that type. |
| -102 | Syntax error | An unrecognized command or data type was encountered. |
| -103 | Invalid separator | The parser was expecting a separator and encountered an illegal character. |
| -104 | Data type error | The parser recognized a data element different than one allowed. |
| -108 | Parameter not allowed | More parameters were received than expected for the header. |
| -109 | Missing parameter | Fewer parameters were received than required for the header. |
| -110 | Command header error | An error was detected in the header. |
| -120 | Numeric data error | Error is generated when parsing a data element which appears to be numeric, including the non-decimal numeric types. |
| -121 | Invalid character in number | An invalid character for the data type being parsed was encountered. |
| -123 | Exponent too large | The magnitude of the exponent was larger than 32000. |
| -124 | Too many digits | The mantissa of a decimal numeric data element contained more than 255 digits excluding leading zeros. |
| -131 | Invalid suffix | The suffix does not follow the syntax or suffix is inappropriate for this device. |
| -141 | Invalid character data | Either the character data element contains an invalid character or the particular element received is not valid for the header. |
| -160 | Block data error | Error is generated when parsing a block data element. |
| -171 | Invalid Expression | The expression data element was invalid; for example, unmatched parentheses or an illegal character. |
| -200 | Execution error | Execution error |
| -213 | Init ignored | Indicates that a request for a measurement initiation was ignored as another measurement was already in progress. |
| -221 | Settings Conflict | The processed parameter could not be set due to the current device state. |

| Error Number | Error String | Description |
|---|---|---|
| -222 | Data Out of Range | The processed parameter is outside of the instrument's defined range. |
| -224 | Illegal parameter value | Indicates that an exact value, from a list of possibles, was expected. |
| -230 | Invalid trace data | Data corrupt or stale |
| -233 | Invalid version | Indicates that the version does not match the version expected by the device. |
| -251 | Missing mass storage | Indicates that a legal program command or query could not be executed because of missing mass storage. |
| -256 | File name not found | Indicates that a legal program command or query could not be executed because the file name on the device was not found. |
| -300 | Device-specific error | Device-specific error |
| -340 | Calibration Failed | Calibration Failed |
| -350 | Queue overflow | A specific code entered into the queue in lieu of the code that caused the error. This code indicates that there is no room in the queue and an error occurred but was not recorded. |
| -363 | Input buffer overrun | Software or hardware input buffer on incoming port overflows with data caused by improper or nonexistent pacing. |
| -400 | Query error | Query error |

# Appendix B — SCPI Command Reference

Anritsu

10580-00481 H

Anritsu utilizes recycled paper and environmentally conscious inks and toner.

Build: March 2025

Anritsu Company
490 Jarvis Drive
Morgan Hill, CA 95037-2809
USA
http://www.anritsu.com