Programming

Remote Control

- Overview
- Making a Measurement
- Reading/Writing Measurement Data
- Saving and Recalling
- Communication with External Devices
- Status Reporting System
- Working with Automatic Test Systems

VBA Programming

- Introduction to VBA Programming
- Operation Basics of the E5052B's VBA
- Controlling the E5052B
- Controlling Peripherals
- User Defined Window

Command Reference

- COM Object Model
- Notational Conventions
- Command Finder

Remote Control

- Overview
- Making a Measurement
- Reading/Writing Measurement Data

- Saving and Recalling
- Communication with External Devices
- Status Reporting System
- Working with Automatic Test Systems

Overview

- Types of remote control system
- GPIB remote control system
- LAN remote control system
- USB Remote Control System
- Sending SCPI command messages
- LXI

Types of remote control system

Depending on the system controller and the interface, you can configure 4 types of remote control system as shown in the table below.

System controller	*Interface*	Overview
External controller (external computer such as PC and workstation)	*GPIB* (talker/listener mode)	System to control the E5052B and other devices connected via GPIB from the external controller. For more information, refer to GPIB remote control system.
	LAN	System to control the E5052B and other devices connected via LAN from the external controller. For more information, refer to LAN remote control system.
	USB	System to control the E5052B and other devices connected via USB

		from the external controller. For more information, refer to USB Remote Control System.
E5052B		System to control the E5052B itself using built-in E5052B VBA.
	GPIB (system controller mode)	System to control the E5052B itself and external devices connected via GPIB using built-in E5052B VBA.

Other topics about Overview

GPIB remote control system

- About GPIB
- System Configuration

Other topics about Overview

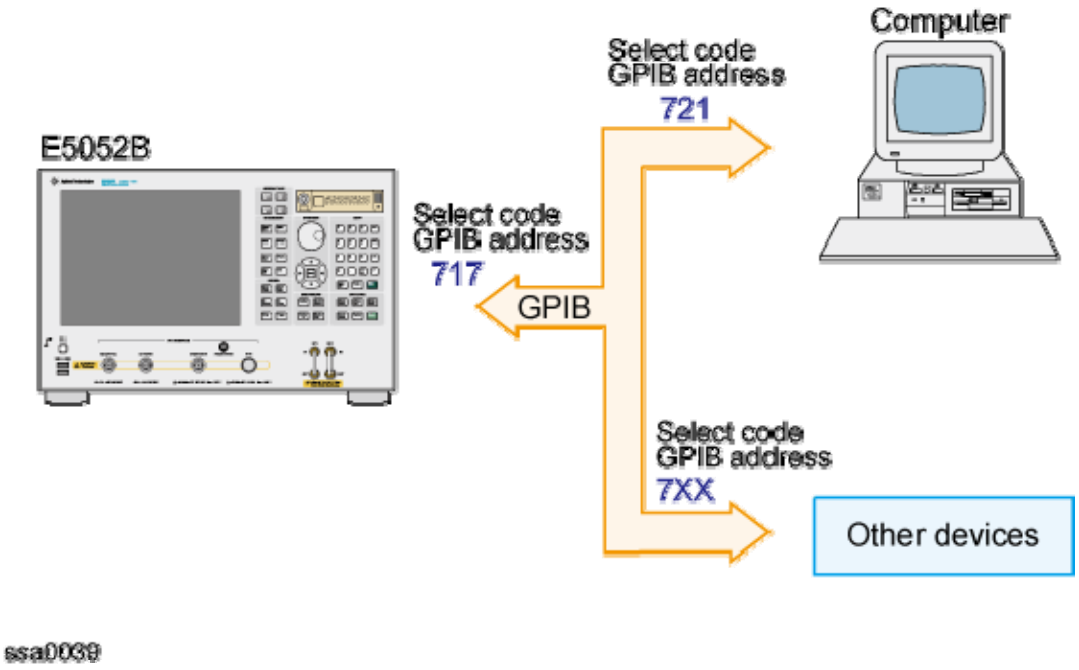
About GPIB

GPIB (General Purpose Interface Bus) is an interface standard for connecting computers and peripherals, which supports the following international standards: IEEE 488.1, IEC-625, IEEE 488.2, and JIS-C1901. The GPIB interface allows you to control the Agilent E5052B from an external computer. The computer sends commands and instructions to the E5052B and receives data sent from the E5052B via GPIB.

System Configuration

Use GPIB cables to connect between the E5052B, the external controller (computer), and peripherals. The following figure shows the overview of the system configuration of the GPIB remote control system.

Configuration of the GPIB remote control system



NOTE

While the E5052B is turned off, the SRQ status of the E5052B is active. To prevent an incorrect operation on the SRQ of the GPIB remote control system, disconnect the E5052B from the system when the E5052B is turned off.

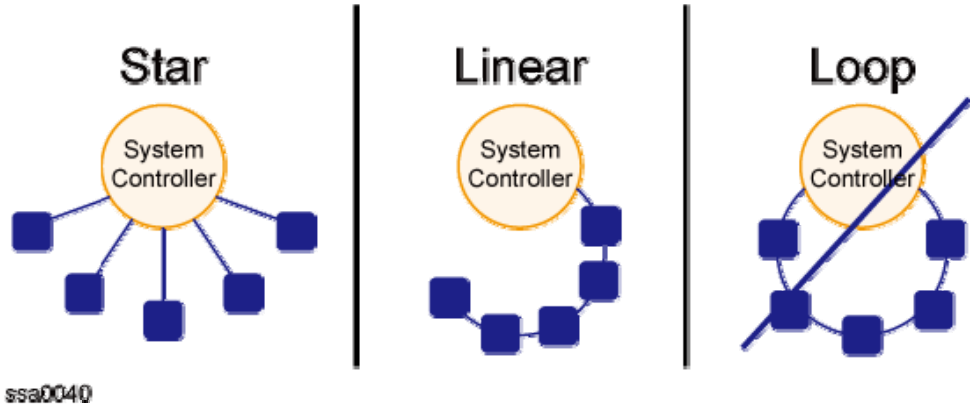
Required Equipment

- E5052B
- External controller (PC or workstation that can be connected to LAN and Agilent I/O Library is installed into)
- Other devices (other instruments and/or peripherals that serve your purpose)
- GPIB cables

Scale of system you can construct

- You can connect up to 15 devices in a single GPIB system.
- The length of cables to connect between devices must be 4 m or less. The total length of connecting cables in a single GPIB system must be 2 m × the number of connected devices (including the controller) or less. You cannot construct the system in which the total cable length exceeds 20 m.

- The number of connectors connected to an individual device must be 4 or less. If you connect 5 or more connectors, excessive force is applied to the connector part, which may result in failure.
- You can choose the device connection topology from star, linear, and combined. Loop connection is not supported.



Device selector

The device selector is a unique value assigned to each device that is used by the controller to select the control target (to send/receive messages) among devices connected on the GPIB remote control system.

The device selector consists of a select code (usually, 7) and a GPIB address. For example, when the select code is 7 and the GPIB address is 17, the device selector is 717. The select code must be set for each system. The GPIB address must be set to a unique value for each device, which is used to identify devices on the same system. In the description and sample programs in this manual, it is assumed that the device selector is set to 717.

Setting the GPIB address of E5052B

To set the GPIB address for talker/listener mode, See Setting_talker_listener_GPIB_address_of_E5052B.

LAN remote control system

- Overview
- System Configuration
- Required Equipment
- Control over SICL-LAN Server

- Control using C or Visual Basic
- Control using Agilent VEE
- Control with Telnet Server

Other topics about Overview

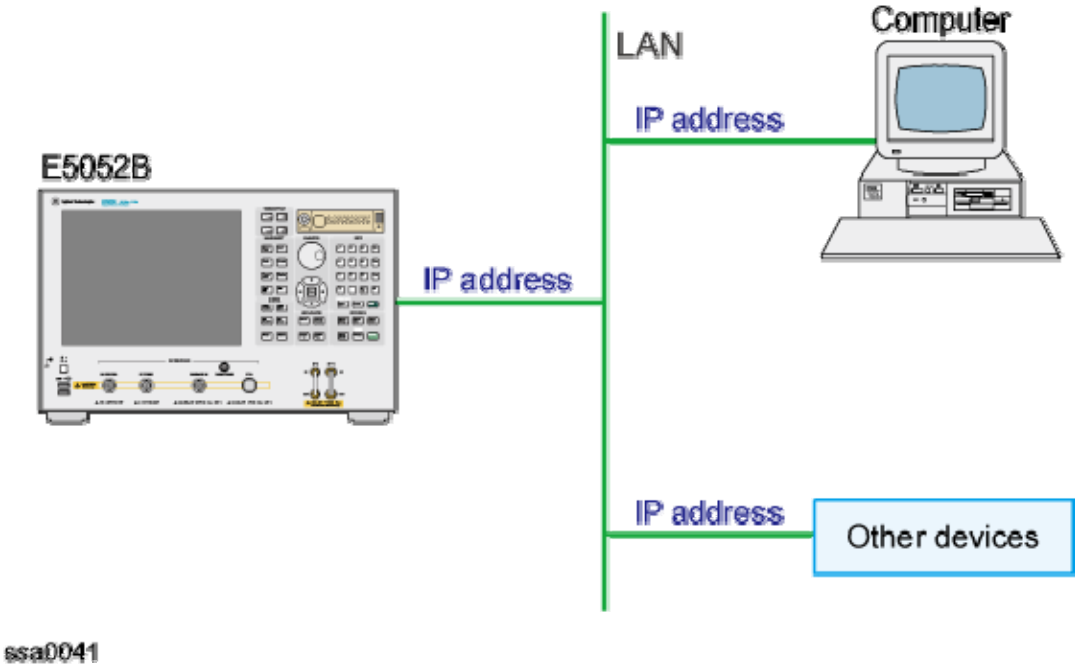
Overview

The LAN (Local Area Network) remote control system provides two methods: controlling the E5052B using the SICL-LAN server and controlling the E5052B using the telnet server.

System Configuration

Use a LAN cable to connect between the E5052B and the external controller (computer). The following figure shows the overview of the system configuration of the LAN remote control system.

Configuration of the LAN remote control system



Required Equipment

- E5052B
- External controller (PC or workstation that can be connected to LAN)

- Other devices (other instruments and/or peripherals that serve your purpose)
- LAN cables

Control over SICL-LAN Server

In the control system using the SICL-LAN server, communication between the external controller (client) and the E5052B (server) is performed using the SICL-LAN protocol. Communication is performed using SICL (Standard Instrument Control Library). You can control the E5052B by programming using SICL or VISA with the C language in the UNIX environment, or Visual C++, Visual Basic, or VEE in the Windows environment.

Preparing the E5052B

To communicate with the external controller, follow these steps to turn on the SICL-LAN server of the E5052B in advance.

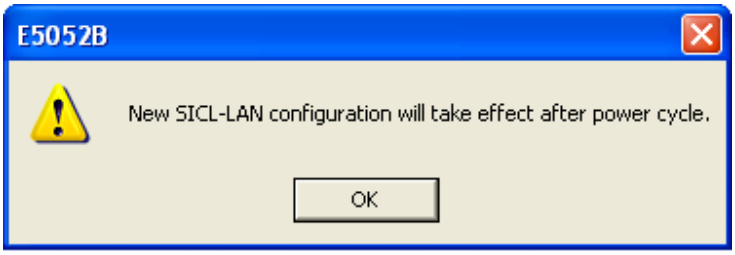
1. Turn on the SICL-LAN server of the E5052B.

   System > **Misc Setup** > **Network Setup** > **SICL-LAN Server [ON]**

2. Set the SICL-LAN address of the E5052B for control with the SICL-LAN server. "XX" represents an address number.

   System > **Misc Setup** > **Network Setup** > **SICL-LAN Address [XX]**

3. By default, the SICL-LAN Address does not changes until the firmware of E5052B is restarted. Restart the firmware to reset the SICL-LAN Address.



ssa0122

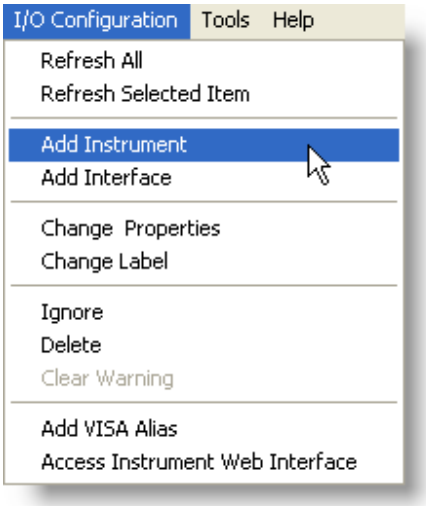
Preparing the external controller

In order to establish communication to the E5052B using the TCP/IP protocol, you need to set the I/O interface of the external controller in advance. This section shows the setting procedure when using the external controller in the Windows environment.

NOTE You must install the Agilent I/O Libraries on your PC in advance. Use Agilent I/O Libraries Suite 14.2 or later.

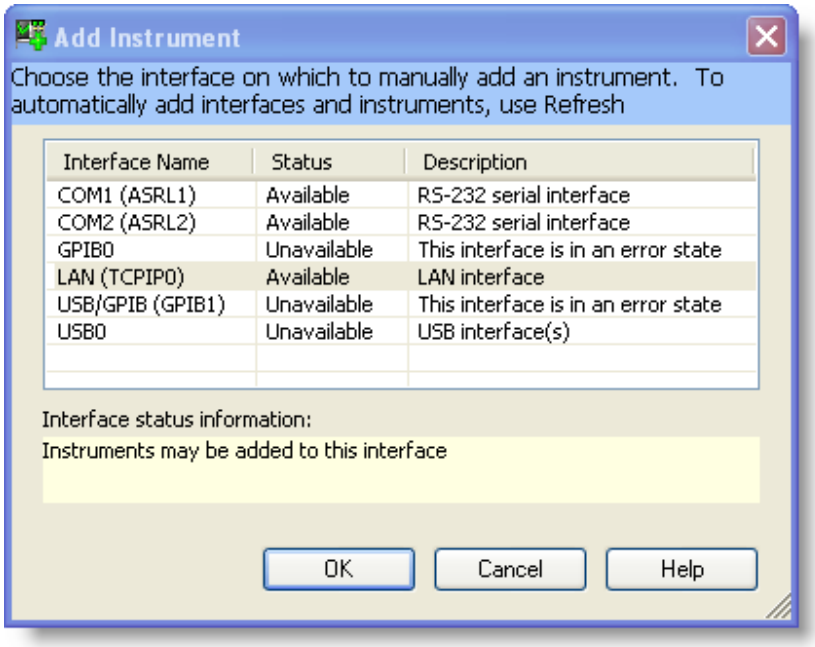
1. From your PC's Start menu, click **Program** > **Agilent I/O Libraries Suite** > **Agilent Connection Expert** to open the Agilent Connection Expert setting screen.

2. In the Agilent Connection Expert setting screen, select **LAN(TCPIP0)** in the **Instrument I/O on this PC** frame, and then click **I/O Configuration** > **Add Instrument.**



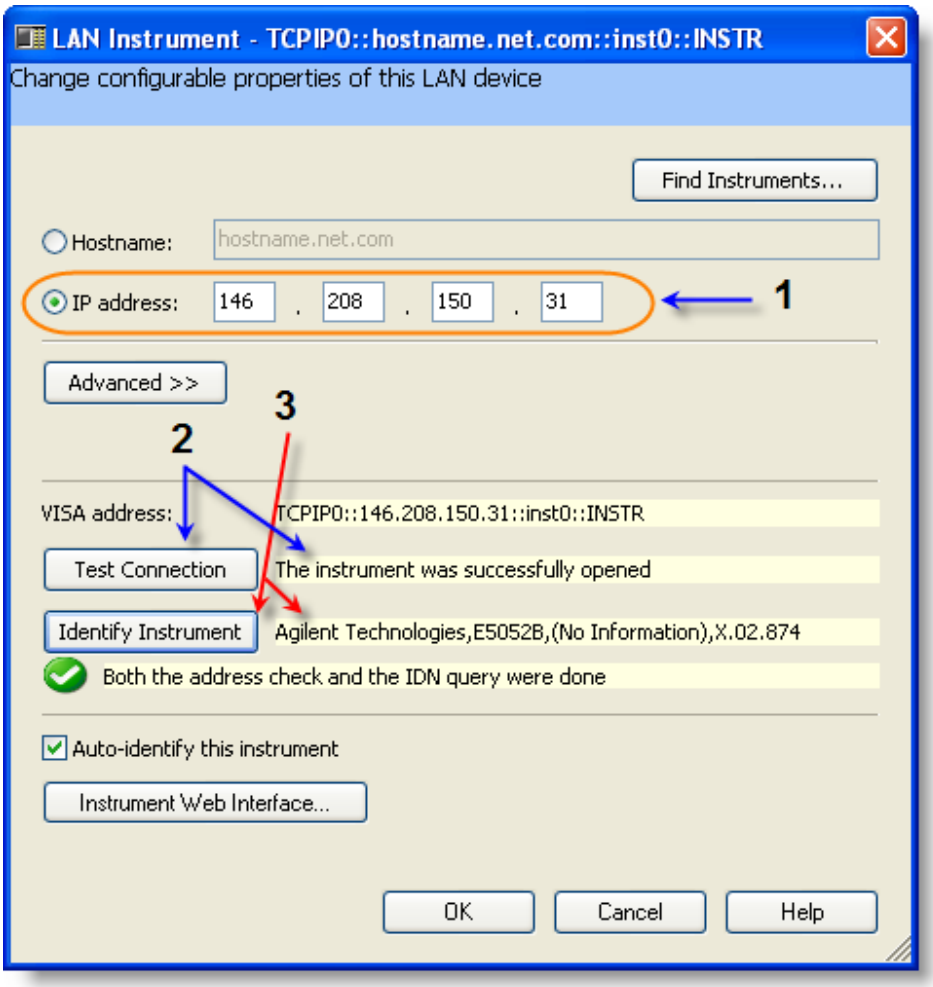
ssa0108

3. In the **Add Instrument** screen, select **LAN** (if it is not selected), and then click **OK**.



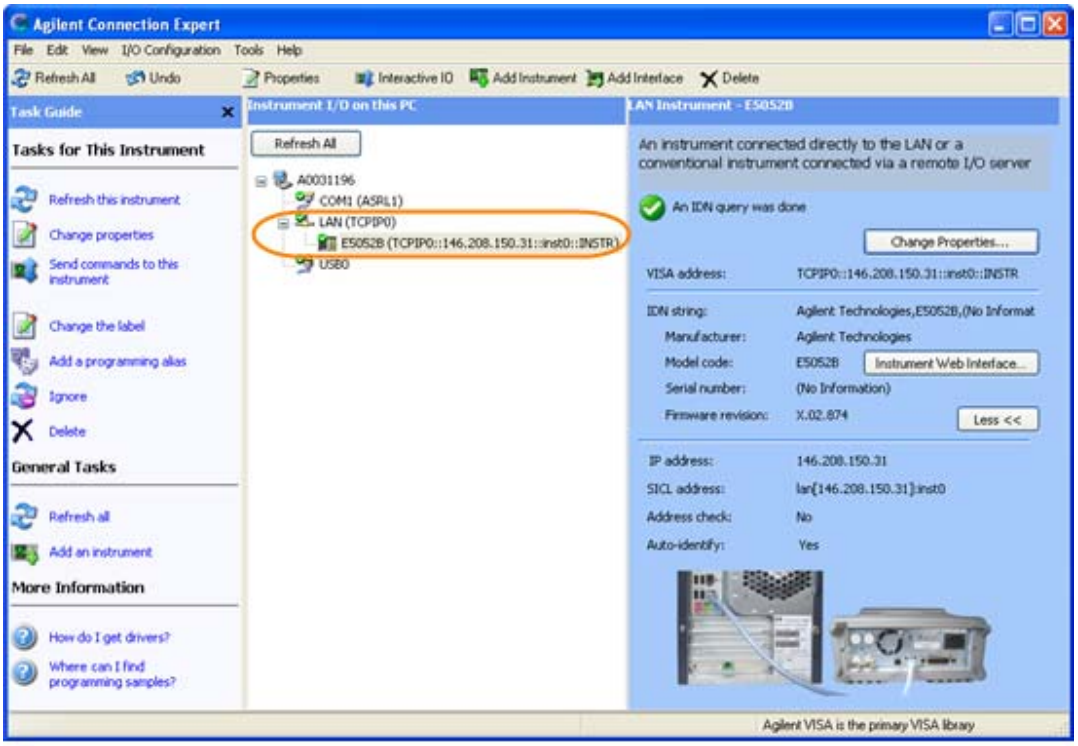
ssa0109

4. In the LAN Instrument Properties screen, set up the IP address of the E5052B (**1** in the figure below) and click **Test Connection** (**2** in the figure below). Click **Identify Instrument** (**3** in the figure below) to identify the E5052B and then click **OK**. You can change settings as necessary. For details, refer to the Agilent I/O Libraries Suite documentation.



ssa0110

5. In the Agilent Connection Expert screen, check that the E5052B has been added under **LAN(TCPIP0)** in the **Instrument I/O on this PC** frame.



ssa0111

Control using C or Visual Basic

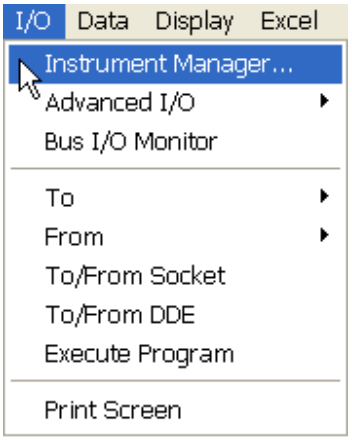
You can control the E5052B by programming using SICL with the C language in the UNIX environment, or Visual C++ or Visual Basic in the Windows environment.

Control using Agilent VEE

Agilent VEE allows you to control the E5052B via the I/O interface. The following example shows how to control the E5052B that is set as follows: the address of the SICL-LAN server is 17 and the IP address is 146.208.116.90.

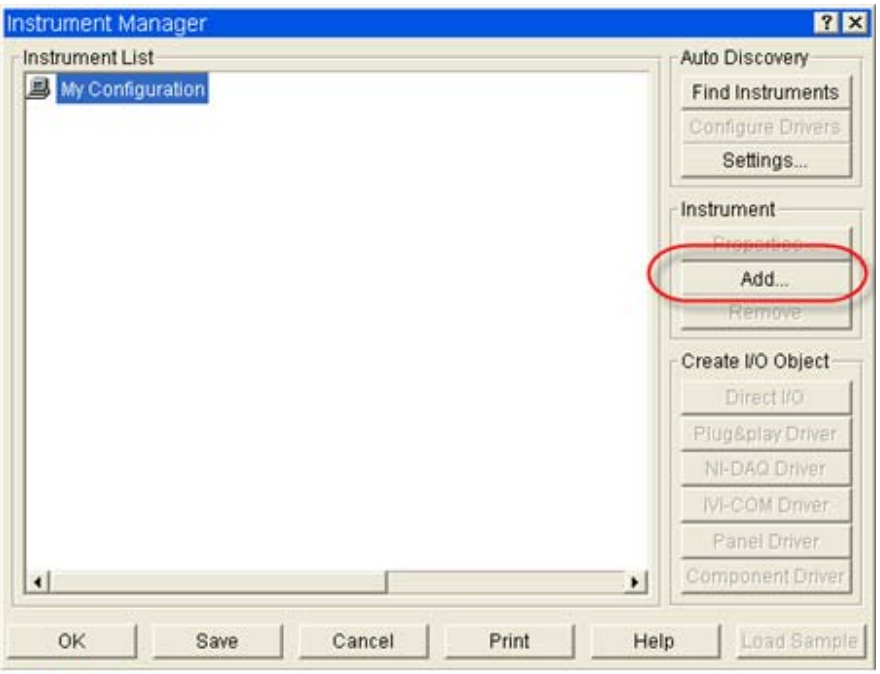
NOTE When using Agilent VEE for PC, use Agilent VEE Pro 7.5 for Windows or later.

1. On the Agilent VEE's **I/O** menu, click **Instrument Manager**



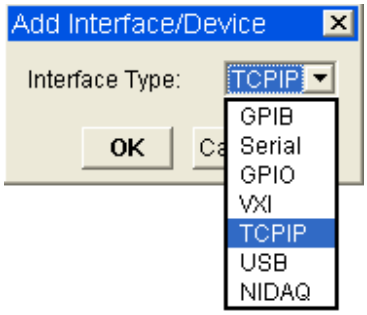
e5071c146

2. In **Instrument Manager**, click **Add...**



e5071c147

3. A new windows appears for the selection of *Interface* Type. Select **TCPIP** and click **OK**.



e5071c148

4. In **Instrument Properties**, type any name for the Instrument in Name (for example: SSA or E5052B), and add TCPIP0::<IP Address> in the **TCIP Address,** where <IP Address> is the IP address for E5052B. For example, if the IP address for E5052B is 146.208.116.90, then the value for **TCPIP Address** would be TCPIP0::146.208.116.90. Click **OK** after entering all the parameters.

5. The Instrument manager displays the connection with E5052B.

Control with Telnet Server

In the control system over telnet server, communications are performed through connection between the sockets provided by the processes of the external controller and the E5052B to establish a network path between them.

A socket is an endpoint for network connection; port 5024 and port 5025 are provided for the sockets for the E5052B. Port 5024 is provided for conversational control using telnet (user interface program for the TELNET protocol) and port 5025 for control from a program.

NOTE To use telnet, port 5024 and 5025 should be opened through Windows firewall.

CAUTION By opening port 5024 and 5025, the E5052B can be controlled remotely using telnet. It is recommended to close port 5024 and 5025 after usage from the security prospective.

Preparing the E5052B

To communicate with the external controller, follow these steps to turn on the telnet server of the E5052B in advance.

System > **Misc Setup** > **Network Setup** > **Telnet Server [ON]**

Conversational control using telnet (using port 5024)

You can use telnet to perform conversational control by sending SCPI commands to the E5052B on a message-by-message basis. For telnet, the socket of port 5024 is used for communications.

In this example, in order to show you the control procedure using telnet, you control the E5052B (IP address: 146.208.116.90 and host name: E5052B) from the external controller in the Windows environment.

1. Open the MS-DOS command prompt screen.
2. At the MS-DOS prompt, type telnet 146.208.116.90 or telnet E5052B and press the return key.
3. The telnet screen opens.
4. Type a command and press the return key; it is sent to the E5052B and executed. If you enter a command that queries some data, the query response is displayed below the line you have entered the command.
5. Press] while holding down Ctl in the telnet screen to break the connection to the E5052B. The telnet prompt appears. At the telnet prompt, type quit and press the *Enter* key. The connection to the E5052B breaks and telnet finishes.

Control from a program (using port 5025)

When controlling the E5052B from a program on the external controller, use the socket of port 5025 for connection.

NOTE Some functions such as service requests that are available in the GPIB remote control system are not available in control over telnet server.

Control using C or Visual Basic

You can control the E5052B by socket programming using the C language in the UNIX environment, or Visual C++ or Visual Basic in the Windows environment.

For socket programming, the library for network connection on the TCP/IP protocol is required. For the UNIX environment, BSD (Berkeley Software Distribution) Sockets API is available; for the Windows environment, WinSock

(WinSock1.1 and WinSock2.0) created by porting BSD Sockets to Windows and expanding it is available.

Control using Agilent VEE

Agilent VEE allows you to control the E5052B through the connection to the socket of port 5025 using To/From Socket. Enter 5025 in **Host Name** to specify the port for connection and enter the IP address or host name of the E5052B in the field to specify the host name.

USB Remote Control System

- Overview
- System Configuration

Other topics about Overview

Overview

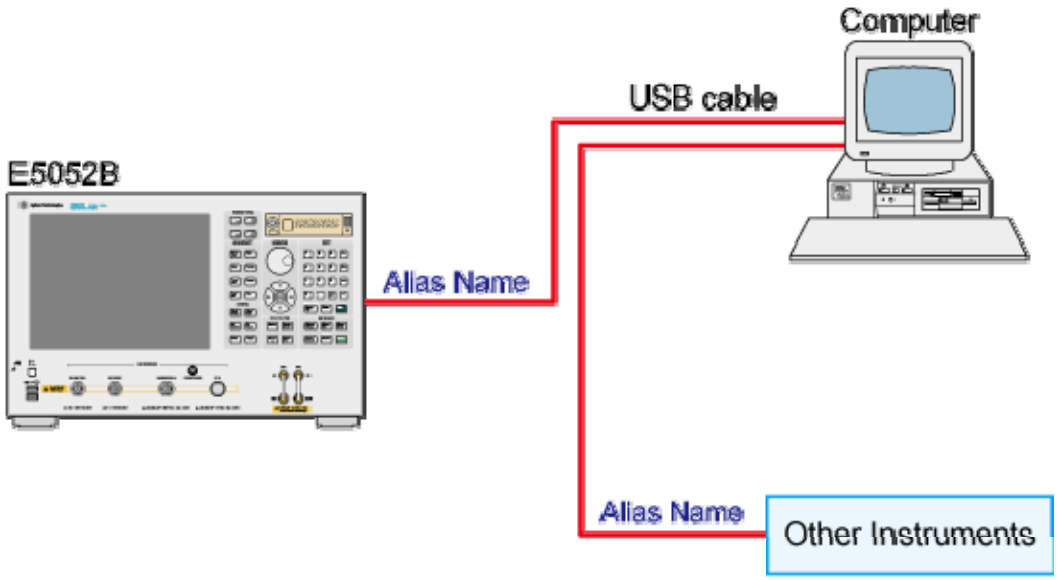
The USB (Universal Serial Bus) remote control system provides device control via USB, which is equivalent to control via *GPIB* . Connection is made through an interface in compliance with USBTMC-USB488 and USB 2.0.

System Configuration

The USB remote control system controls instruments that use the name "alias." There is no such address for GPIB connections.

Use a USB cable to connect the E5052B to an external controller (personal computer). The following figure shows an overview of the system configuration for the USB remote control system.

USB Remote Control System Configuration



ssa0042

Required Equipment

- E5052B
- External controller (PC with USB host port (type A)).
- Other USB compatible devices (instruments and/or peripherals for specific purposes).
- USB cable connecting E5052B and external controller (with type A/4-prong male or type B/4-prong male connectors depending on device used).

USB Port Types

There are two standard types of USB ports. The external controller (PC) must be connected via the USB host port (type A), while the E5052B and other USB compatible devices must be connected via the USB interface port (type B).

Port Type	Description
	Type A: USB host port

	Type B: USB (USBTMC) interface port

Preparing E5052B

You do not have to configure any softkey or command of the E5052B in order to control the E5052B from an external controller. Simply connect a USB cable to the USB interface port.

Preparing External Controller

In order to establish communication with the E5052B via USB, you must set up the *I/O* interface of the external controller in advance. The USB can identify devices automatically, so once you connect a USB cable to a target device, a dialog box will appear for USB device registration.

NOTE The E5052B will be identified as new device if its serial number has been changed.

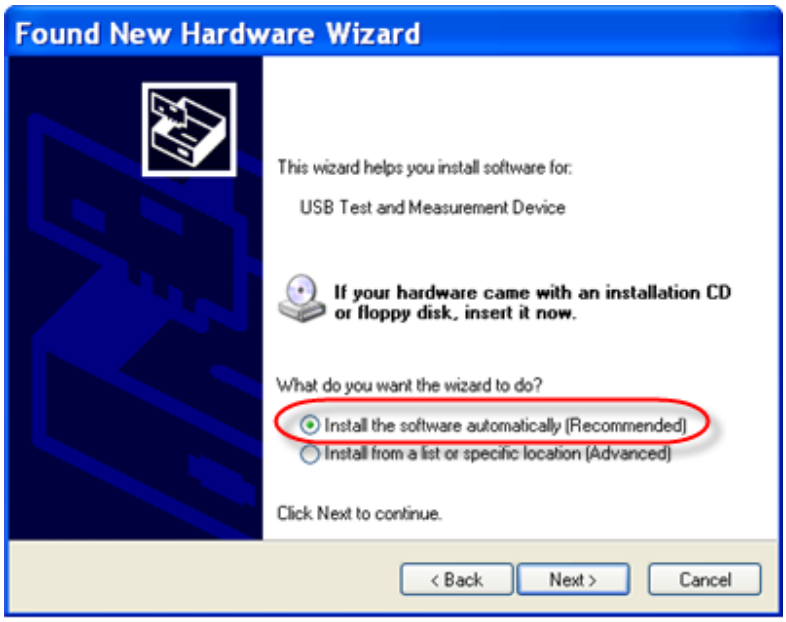
NOTE You must install the Agilent I/O Libraries on your PC in advance. Use Agilent I/O Libraries Suite 14.2 or later.

1. Setting E5052B when USB Cable Is Connected

1. When new device is connected via USB cable, the following dialog box will appear automatically. Select **No, not this time**, and then click **Next**.

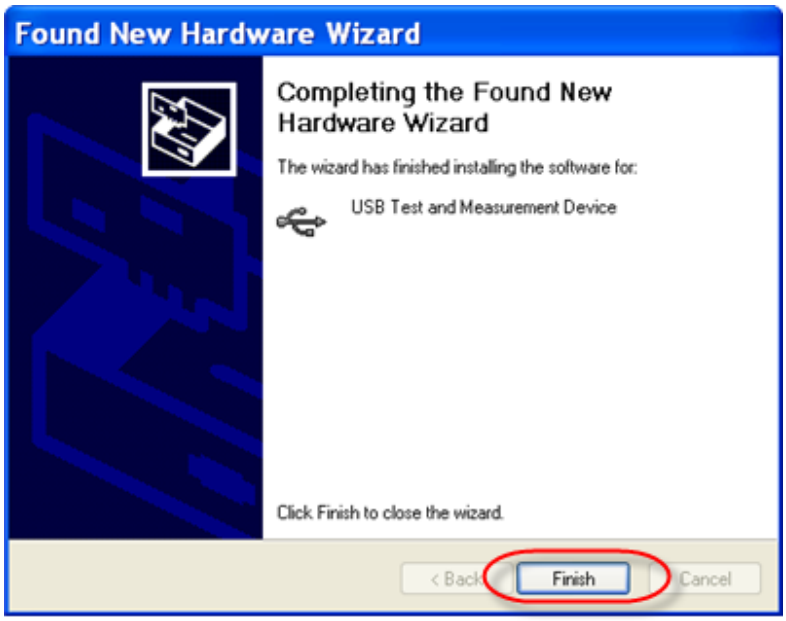
e5071c155

2. Select **Install the software automatically (Recommended)**, and then click **Next**.



e5071c156

3. The drivers for E5052B are automatically installed and the completion screen appears. Click **Finish** to complete the process.



e5071c157

2. Registering Alias

Just after finishing the setting, another screen appears that can be used to change the Alias for the connected E5052B.

NOTE For alias, use the ASCII format less than 127 digits. Alias is upper/lower case insensitive.

NOTE If **Never show this dialog** is selected in **Show this dialog** frame, the dialog box does not appear even if a new device is connected.

NOTE Once new device is identified, the "New Hardware Search Wizard" will start. Follow the instruction to implement the processing.

3. Changing Alias on Setting Screen

The following are steps using the Agilent I/O Libraries Suite 14.2.

1. From the Start menu of your PC, click **Programs** > **Agilent IO Libraries Suite** > **Agilent Connection Expert** to open the Config setting screen.

2. In the Config setting screen, select the alias names from **USB0** onward in the **Instrument I/O on this PC** frame, and then use the **Change Properties** from **I/O Configuration** on the menu bar.

Control using C or Visual Basic

You can control the E5052B by programming using Visual C++ or Visual Basic in the Windows environment as well as SICL/VISA.For further information on controlling the E5052B, see the manual of SICL or VISA. For Agilent I/O Libraries, use Agilent I/O Libraries Suite 14.2 or later.

You may use alias in the programming using SICL/VISA.

The following example shows an OPEN command to control the E5052B to which alias is given as SSA_USBIF.

SICL	id = iopen("SSA_USBIF")
VISA	viOpen(...,"SSA_USBIF",...)

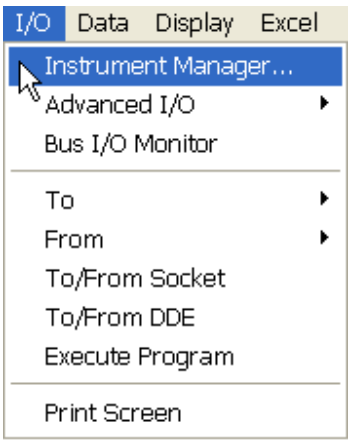
NOTE For further details of the programming using SICL/VISA, see the SICL Users Guide or the VISA Users Guide.

Control using Agilent VEE

Agilent VEE allows you to control the E5052B via the direct I/O interface. The following example shows how to control the E5052B to which alias is given as SSA_USBIF.

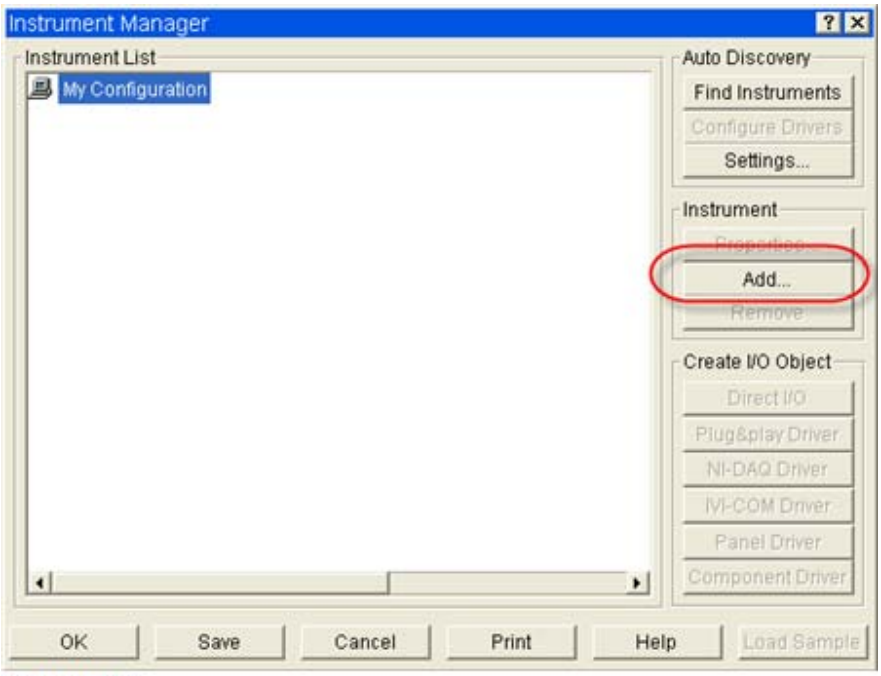
NOTE When using Agilent VEE for PC, use Agilent VEE Pro 7 for Windows or later version.

1. On the Agilent VEE's **I/O** menu, click **Instrument Manager**.



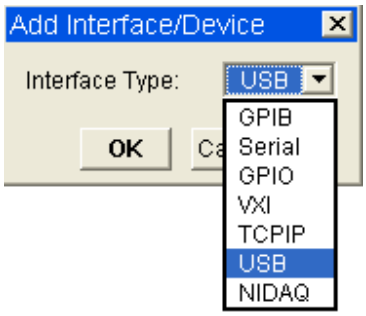
e5071c146

2. In **Instrument Manager**, click **Add...**



e5071c147

3. A new windows appears for the selection of *Interface* Type. Select **USB** and click **OK**.



e5071c154

4. In **Instrument Properties**, type any name for the Instrument in Name (for example: SSA_USBIF or E5052B_USB), and add USB Address in the **USB Address.** Click **OK** after entering all the parameters.

NOTE The USB address can be retrieved from Agilent Connection Expert.

5. The E5052B successfully appears in the **Instrument Manager**.

Sending SCPI command messages

- Type and Structure of Commands
- Grammar of Messages
- Remote Mode

Other topics about Overview

Type and Structure of Commands

The SCPI commands available for the E5052B are classified into 2 groups as follows.

E5052B commands

Commands specific to the E5052B. They cover all measurement functions that the E5052B has and some general-purpose functions. The commands in this group are arranged in a hierarchical structure called the command tree. Each command consists of character strings (mnemonics) indicating each hierarchical level and colon (:) separators between hierarchical levels.

IEEE common commands

Commands to cover general-purpose functions defined in IEEE488.2 that are available commonly to instruments that support this standard. The commands in this group have an asterisk (*) at the beginning. For the commands in this group, there is no hierarchical structure.

Concepts of the command tree

The commands at the top of the command tree are called "root command" or simply "root." To access lower level commands in the tree, you need to specify a specific path like a directory path in the DOS file system. After power-on or reset, the current path is set to the root. Special characters in messages change the path setting as described below.

Message terminator

A message terminator such as the
<new line> character sets the current path to the root.

Colon (:)

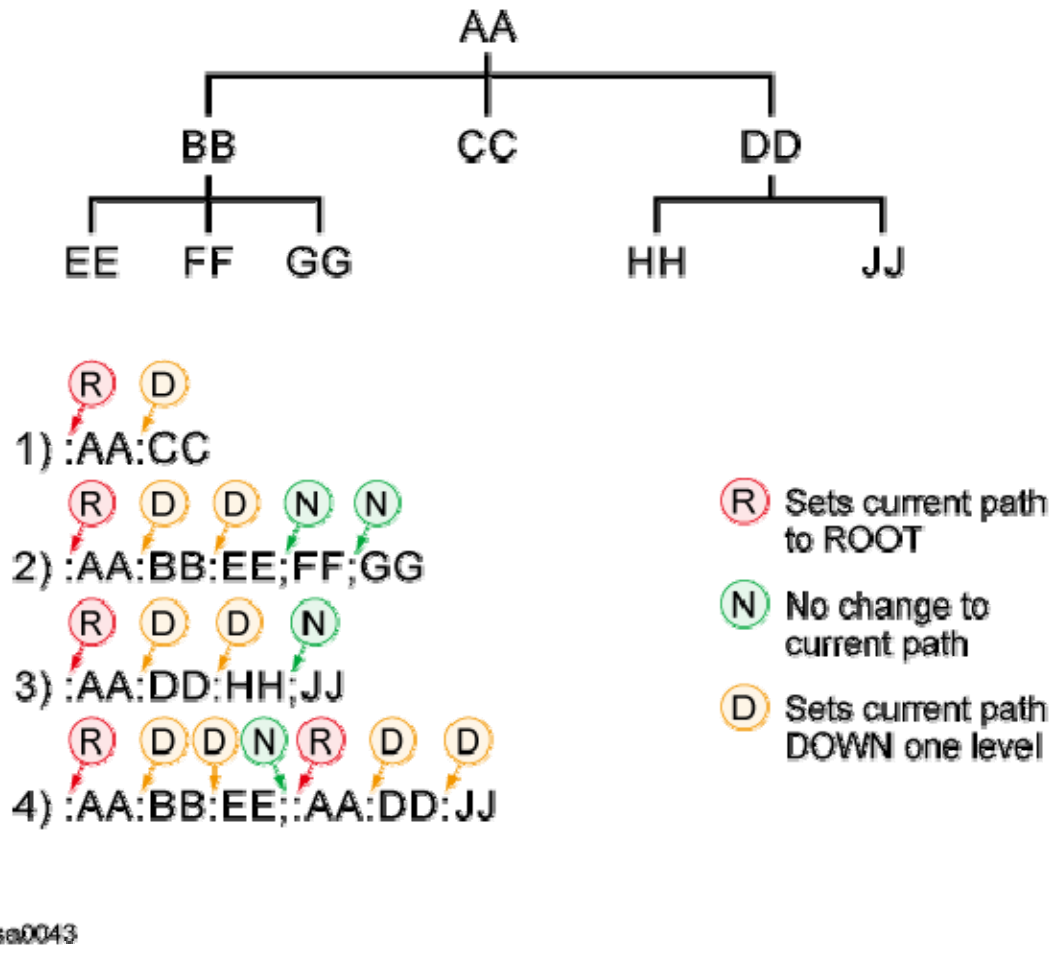
A colon between 2 command mnemonics lowers the level of the current path in the command tree. A colon used as the first character of a command specifies the command mnemonic that follows as the root-level command.

Semicolon (;)

A semicolon does not change the current path and separates 2 commands in the same message.

The following figure shows an example of how to use colons and semicolons to efficiently access commands in the command tree.

Using colons and semicolons



Grammar of Messages

This section describes the grammar to send program messages via *GPIB* . Program messages are messages that the user sends to the instrument from the

external controller to control the instrument. A program message consists of 1 or more commands and their necessary parameters.

Upper/lower case sensitivity

Upper/lower case insensitive.

Program message terminator

A program message must be terminated with one of the 3 program message terminators: <new line>, <^END>, or <new line><^END>. <^END> indicates that EOI on the GPIB interface becomes active at the instant when the immediately previous data byte is sent. For example, the OUTPUT command of HTBasic automatically sends the message terminator after the last data byte.

Parameters

A space (ASCII code: 32) is required between a command and its first parameter. When sending several parameters in a single command, separate each parameter with a comma (,).

Message including several commands

When sending 2 or more commands in a single message, separate each command with a semicolon (;). The following example shows how to send the *CLS command and the :STAT:PRES command in a single message using HTBasic.

```
OUTPUT 717;"*CLS;:STAT:PRES"
```

Remote Mode

The E5052B does not provide remote mode. Therefore, even if you send a GPIB command, it never enters into remote mode automatically. There is no local key to release remote mode.

If you need to prevent misoperation during remote control due to entry from the front panel or mouse, lock the input devices using the following commands:

- :SYST:KLOC:KBD
- :SYST:KLOC:MOUS

LXI

The E5052B is LXI-C compliant from revision A.03.10 onwards.

About LXI

LXI (LAN eXtensions for Instrumentation) is the LAN-based successor to GPIB and combines the advantages of Ethernet with the simplicity and familiarity of GPIB. The key features of LXI are as follows:

- The speed, simplicity, worldwide reach, low cost, ongoing enhancement and backward compatibility of LAN.
- Quick, easy configuration through the intuitive web interface built into compliant instruments.
- Simplified programming and greater software reuse through IVI drivers.
- The ability to create hybrid systems that include LXI, GPIB, VXI, PXI, CANbus, etc.
- Enhanced system performance and event handling via hardware- and LAN-based triggering modes.
- Synchronization of local and remote instruments through the IEEE 1588 precision time protocol.

NOTE For more information on LXI, please visit www.lxistandard.org

Checking LXI Compliance

The E5052B having LXI compliance will show a LXI logo at the start up screen of the firmware. If this logo does not appears at the start up screen of the firmware, it means that the system is not LXI-C complaint.

Files Required for LXI for E5052B with HDD revision NM30x

The HDD revision NM303 or below does not comes with installed Dot Net architecture which is a requirement for LXI. User having HDD revision NM303 or below need to install the required components for LXI on the E5052B, and then update the firmware to A.03.10 or later to use LXI-C functionality.

LXI works only on Dot Net platform. To install dotnet framework and other software required for LXI:

1. Exit E5052B by **System** > **Service Menu** > **Administrator Menu** > **Exit E5052B**.
2. Install ".Net Framework 2.0" by downloading the file **dotnetfx.exe** from http://www.agilent.com/find/ssa, save it to User folder, and then execute it to install all the required components for LXI on the E5052B.
3. If your E5052B firmware revision is A03.03 or below, update E5052B Firmware to A.03.10 or later.
4. Enable Web Server by **System** > **Misc Setup**> **Network Setup** > **Web Server [ON]**.

Making Measurement

- Trigger System
- Staring a Measurement Cycle (triggering the instrument)
- Waiting for End of measurement

Trigger System

- Overview
- Trigger state and transition for each measurement

Other topics about Making Measurement

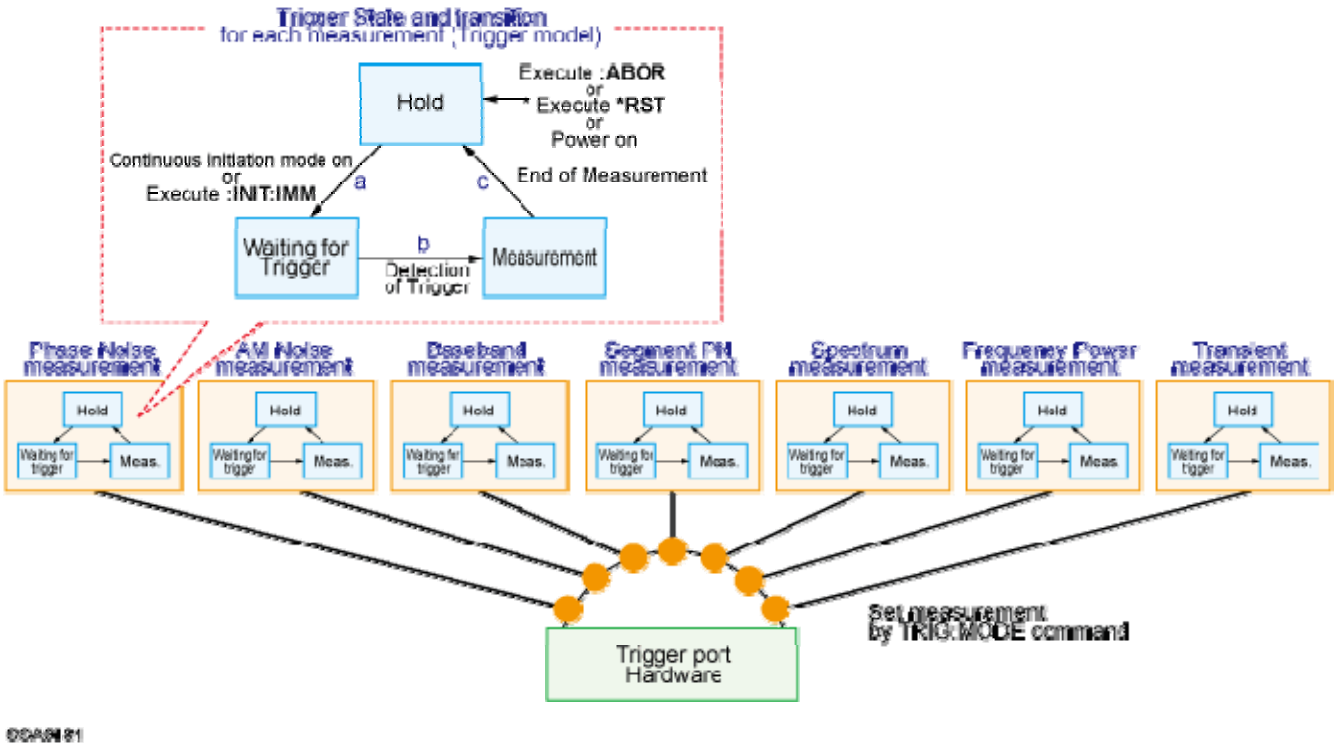
Overview

The trigger system is responsible for tasks such as detecting the start of a measurement cycle (triggering) and enabling/disabling measurement for each measurement. As shown in the following figure, the trigger system has three states for each measurement: “Hold,” “Waiting for *Trigger* ,” and “Measurement”. If any one of the measurements holds hardware and trigger port, other measurement will keep one of the states.

NOTE

The E5052B has a trigger model for each measurement (see below); however, only one triggered measurement can take place at a time. This is because the hardware and the trigger port are shared by seven measurement interfaces. For example, even if the continuous initiation mode is enabled for all measurements and the trigger source is set to Internal, only one triggered measurement takes place.

Trigger system



NOTE When the windows couple is turned on, you cannot use TRIG:MODE.

Trigger state and transition for each measurement

The following subsections describe the states for each measurement and how the trigger system switches among the states.

“Hold” State

When one of the following commands has been executed, the trigger system switches to “Hold” state, interrupting the measurement which is underway (arrow “e” in “d” in the Trigger System). When the power is turned on, the phase noise measurement is triggered, while the continuous initiation mode is set to ‘ON’ for the phase noise measurement and the trigger source is set to “Internal”.

- :ABORt
- *RST

During the “Hold” state, if either one of the following commands is executed or the measurement switches to the “Initiate” state by the front panel control, then the trigger system switches to the “Waiting for Trigger” state (arrow “f” in “a” in

the Trigger System). “xx” specifies the measurements (PN, AM, BB, PS, SP, FP or TR) that are now waiting for trigger.

- :INITiate:xx:IMMediate
- :INITiate:xx:CONTinuous ON

“Waiting for Trigger” State

The instrument is triggered (i.e., a trigger is detected) during the “Waiting for Trigger” state, and then the trigger system switches to the “Measurement” state (arrow “B” in “b” in the Trigger System).

As shown in the table below, how the instrument is triggered differs depending on which trigger source is specified.

Trigger Source	How instrument is triggered
Internal trigger	The instrument is automatically triggered itself.
External trigger	The instrument is triggered when a trigger signal is fed through the Ext Trig terminal.
Bus trigger	The instrument is triggered when the *TRG command is issued.
Manual trigger	The instrument is triggered when you press [Trigger] - Trigger on the front panel. **Trigger** > **Manual Trigger**
Wide band trigger	In using the wideband mode for the transient measurement, the instrument is triggered when the frequency of the measuring signal crosses the frequency value of the wideband trigger setting.
Narrow band trigger	In using the narrowband mode for the transient measurement, the instrument is triggered when the frequency of the measuring signal crosses the frequency value of the narrowband trigger setting.

To set the trigger source, use the following command (“xx” specifies the measurements (PN,AM,BB,PS,SP,FP,TR) that are now waiting for trigger):

- :TRIGger:xx:SOURce

“Measurement” State

In the “Measurement” state, the instrument starts the measurement that was in the “Initiate” state immediately before the transition to this state.

If the delay time of DC control voltage and DC power voltage are set (by :SOURce:VOLTage:CONTrol:DELay, :SOURce:VOLTage:POWer:DELay), the instrument waits for the elapse of the sweep delay time before starting a measurement.

When the instrument has finished measuring, the trigger system behaves in one of the following ways depending on the setting of the continuous initiation mode.

Continuous Initiation Mode	Trigger System Behavior
OFF	The trigger system switches to the “Hold” state; “c” in the Trigger System
ON	The trigger system switches to the “Hold” state and then to the “Waiting for Trigger” state; “c” and “a” in the Trigger System

Staring a Measurement Cycle (triggering the instrument)

- Configuring the Instrument to Automatically Perform Continuous Measurement
- Starting Measurement on Demand
- Using the Averaging Trigger

Other topics about Making Measurement

Configuring the Instrument to Automatically Perform Continuous Measurement

1. Select a measurement to be triggered using the :TRIGger:MODE command.
2. Set the trigger source of the triggered measurement to Internal using the :TRIGger:xx:SOURce command.

3. Set the continuous initiation mode of the enabled triggered measurement using the :INITiate:xx:CONTinuous command.

Starting Measurement on Demand

1. Select a measurement to be triggered using the :TRIGger:MODE command.
2. Set the trigger source of the triggered measurement to Bus using the :TRIGger:xx:SOURce command.
3. Set the continuous initiation mode of the enabled triggered measurement using the :INITiate:xx:CONTinuous command.
4. Trigger the instrument when you want to perform measurement. An external controller can trigger the instrument by only a single command.

Command	Applicable trigger source
*TRG	Bus trigger only

5. To start the next measurement cycle, repeat step 4.

Using the Averaging Trigger

You can make the E5052B repeat a cycle of waiting for a trigger and performing measurement for the number of times specified by the averaging factor with a single transition to the “Initiate” state. To switch between the averaging trigger mode and the normal trigger mode, use the following command.

- :TRIGger:AVERage

Averaging trigger mode	State transition for each setting
On	Turns on the averaging trigger mode. The instrument repeats a cycle of "waiting for a trigger" to "performing measurement," as many times as specified by the averaging factor, and then transitions to the “Hold” state.
Off	Turns off the averaging trigger mode. When the instrument finishes a single cycle of “waiting for a trigger” to “performing measurement,” it transitions to “Hold” state.

The following table shows the operational difference between the trigger systems.

Continuous initiation mode	Averaging trigger mode	Operation of trigger system
Off (Single)	Off	The instrument waits for a trigger only once. When it receives a trigger, it performs measurement and then transistors to the “Hold” state. The averaging factor is not cleared; it is incremented.
	On	The instrument repeats a cycle of “waiting for a trigger” to “receiving a trigger” to “performing measurement,” as many times as specified by the averaging factor, and then transitions to the “Hold” state. The averaging factor is cleared before the start of measurement after receiving the first trigger.
On (Continuous)	Off	The instrument repeats a cycle of “waiting for a trigger” to “receiving a trigger” to “performing measurement.” The averaging factor is not cleared; it is incremented.
	On	The instrument repeats a cycle of “waiting for a trigger” to “receiving a trigger” to “performing measurement.” The averaging factor is cleared before the start of measurement after receiving the first trigger.

NOTE The setting made with this command is common to all of the measurement modes (PN, PS, SP, FP, AM, BB and TR). You cannot use a different setting for each measurement mode.

NOTE When the averaging mode (set with the :SENS:xx:AVER:STAT command) is off, the averaging trigger mode is not available.

Waiting for End of measurement

- Using the Status Register
- Using the *OPC command
- Using Wait Time

Other topics about Making Measurement

Using the Status Register

The status of the E5052B can be detected through the status registers. This section describes how to detect the end of measurement by using the status registers. For a complete description of the status report mechanism, including the specifications of each bit, see “Status Reporting System,”.

Measurement status is reported by the operation status condition register (see Opeartion Status Condition Register). An SRQ (service request) is useful when you create a program that uses the information reported by this register to detect the end of measurement.

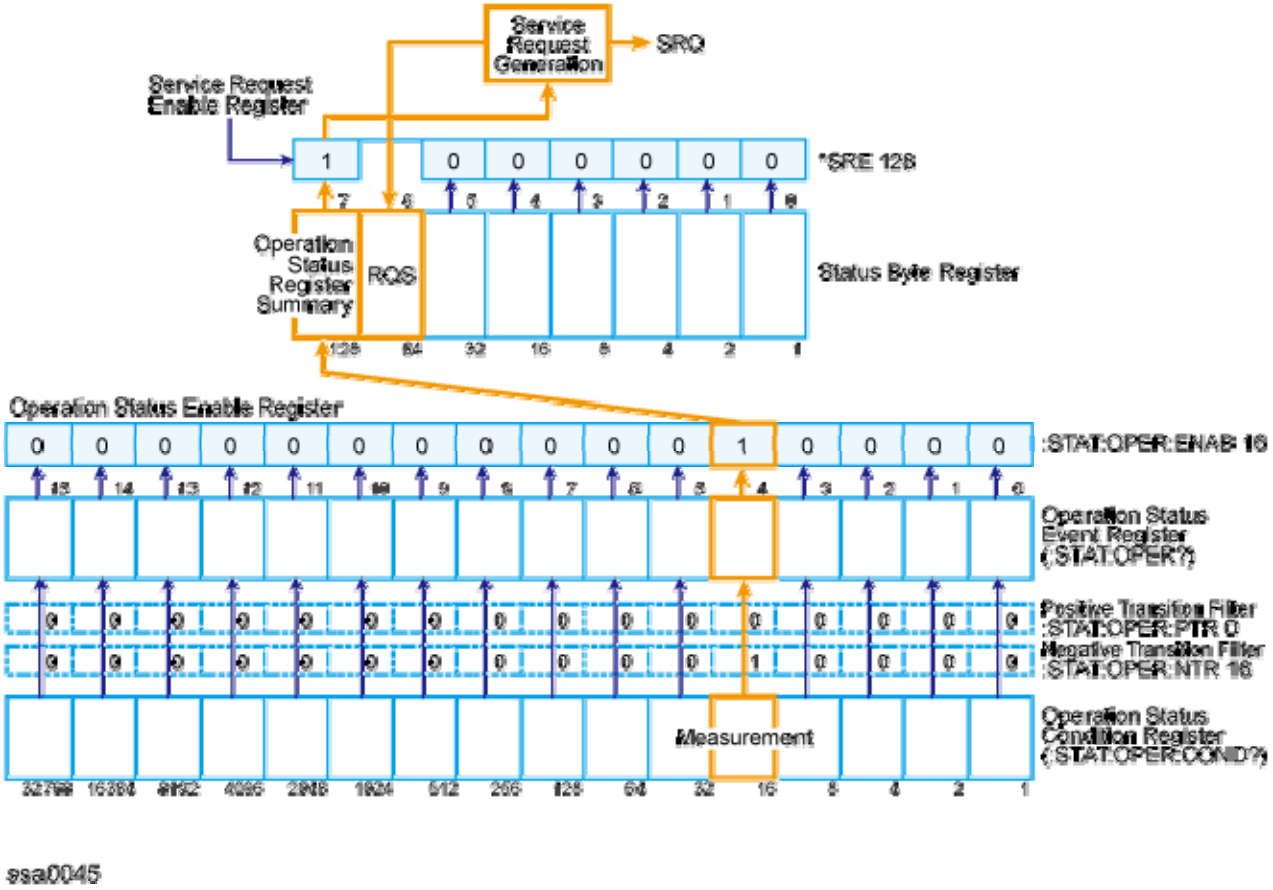
To detect the end of measurement via an SRQ, use one of the following commands:

- *SRE
- :STATus:OPERation:ENABle
- :STATus:OPERation:PTRansition
- :STATus:OPERation:NTRansition

Follow these steps:

1. Configure the E5052B so that it generates an SRQ when the operation status condition register's bit 4 (a bit that is set to 1 during measurement) is changed from 1 to 0.
2. *Trigger* the instrument to start a measurement cycle.
3. When an SRQ is generated, the program interrupts the measurement cycle.

SRQ generation sequence (at end of measurement)



Sample program

The following is a sample program that demonstrates how to use an SRQ to interrupt the end of measurement.

This program triggers the spectrum measurement of E5052B and then terminates the program when the end signal is received.

Sample program showing how to interrupt the end of measurement

```
110 DIM Buff$[9]
120 INTEGER A
130 ASSIGN @Agte5052 TO 717
140 OUTPUT @Agte5052;":ABOR"
150 OUTPUT @Agte5052;":TRIG:MODE SP1"
160 OUTPUT @Agte5052;":TRIG:SP:SOUR BUS"
170 OUTPUT @Agte5052;":INIT:SP:CONT ON"
```

```
180 OUTPUT @Agte5052;":STAT:OPER:PTR 0"
190 OUTPUT @Agte5052;":STAT:OPER:NTR 16"
200 OUTPUT @Agte5052;":STAT:OPER:ENAB 16"
210 OUTPUT @Agte5052;"*SRE 128"
220 OUTPUT @Agte5052;"*CLS"
230 OUTPUT @Agte5052;"*OPC?"
240 ENTER @Agte5052;Buff$
250 ON INTR 7 GOTO Meas_end
260 ENABLE INTR 7;2
270 OUTPUT @Agte5052;"*TRG"
280 PRINT "Waiting..."
290 Meas_wait: WAIT .1
300 OUTPUT @Agte5052;"*STB?"
310 ENTER @Agte5052;A
320 PRINT "a=";A
330 GOTO Meas_wait
340 Meas_end: OFF INTR 7
350 PRINT "Measurement Complete"
360 END
```

Using the *OPC? command

You can detect end of measurement by executing the *OPC? command or the *WAI command in your program. This section describes how to detect the end of measurement using the *OPC? command.

To enable the *OPC?, *WAI, and *OPC commands, you need to turn on these commands with the :TRIGger:SOPC.

- :TRIGger:SOPC

NOTE The setting made with this command is common to all of the measurement modes (PN, PS, SP, FP, AM, BB and TR). You cannot use a different setting for each measurement mode.

The following commands work differently when the :TRIG:SOPC command is turned on.

Com-mand	Operation when :TRIG:SOPC is on	Operation when :TRIG:SOPC is off
*OPC?	The processing of the program is suspended until the end of the measurement, and +1 is returned after the end of the measurement.	The control is returned to the program immediately without waiting for the end of the measurement.
*WAI	The processing of the program is suspended until the end of the measurement.	The control is returned to the program immediately without waiting for the end of the measurement.
*OPC	After the command is executed, the control is returned to the program without waiting for the end of the measurement. The Operation Completion bit of the Standard Event Status register is set after the end of measurement.	The control is returned to the program immediately without waiting for the end of the measurement. The Operation Completion bit of the Standard Event Status register is set without waiting for the end of measurement.

The processing of the program is suspended only during a transition period from “Waiting for Trigger” state to “Hold” state caused by the end of measurement or abortion of the measurement.

When the averaging trigger is on, the processing of the program is suspended until the “Hold” state occurs after all measurements for the averaging factor are completed.

NOTE

When the averaging trigger is on and the continuous initiation mode is on (Continuous), the processing of the program is suspended until the “Hold” state occurs after all measurements for the averaging factor are completed. However, in this case, once the measurements for the averaging factor are completed, the control is returned to the program each time a single measurement is completed for subsequent *OPC? and *WAI commands.

Using Wait Time

Before creating your program, actually measure the time between the start and end of the measurement cycle. Then code your program so that the controller waits for the actually measured time by using the appropriate command (for example, the WAIT command for HTBasic). This is a straightforward method, but care must be taken: an incorrect wait time could result in unexpected error.

Reading Writing Measurement Data

- Data Types for Data Transfer
- Data Transfer Format
- Entering Data in a Trace
- Internal data processing
- Retrieving Measurement Results

Data Types for Data Transfer

- Overview
- Array Type
- Block Type

Other topics about Reading-Writing Measurement Data

Overview

This section describes the data types that the E5052B uses for data transfer.

Array Type

If you use any of the following commands, the data transfer will be executed using array-type data. Here, you can choose from among ASCII transfer format, IEEE 64-bit floating point binary transfer format, and IEEE 32-bit floating point binary transfer format.

For phase noise measurement

- :CALCulate:PN[1-1]:DATA:CARRier
- :CALCulate:PN[1-1]:DATA:PDATa
- :CALCulate:PN[1-1]:DATA:RDATa
- :CALCulate:PN[1-1]:DATA:XDATa
- :CALCulate:PN[1-1]:TRACe[1-1]:DATA:FDATa
- :CALCulate:PN[1-1]:TRACe[1-1]:DATA:FMEMory

- :CALCulate:PN[1-1]:TRACe[1-1]:DATA:PDATa
- :CALCulate:PN[1-1]:TRACe[1-1]:DATA:PMEMory
- :CALCulate:PN[1-1]:TRACe[1-1]:DATA:SDATa
- :CALCulate:PN[1-1]:TRACe[1-1]:DATA:SMEMory
- :CALCulate:PN[1-1]:TRACe[1-1]:DATA:UDATa
- :CALCulate:PN[1-1]:TRACe[1-1]:DATA:UMEMory
- :CALCulate:PN[1-1]:TRACe[1-1]:LIMit:LOWer:LDATa
- :CALCulate:PN[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:DATA
- :CALCulate:PN[1-1]:TRACe[1-1]:LIMit:REPort[:DATA]
- :CALCulate:PN[1-1]:TRACe[1-1]:LIMit:UPPer:LDATa
- :CALCulate:PN[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:DATA
- :CALCulate:PN[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:DATA

For spectrum measurement

- :CALCulate:SP[1-1]:DATA:RDATa
- :CALCulate:SP[1-1]:DATA:XDATa
- :CALCulate:SP[1-1]:TRACe[1-1]:DATA:FDATa
- :CALCulate:SP[1-1]:TRACe[1-1]:DATA:FMEMory
- :CALCulate:SP[1-1]:TRACe[1-1]:DATA:UDATa
- :CALCulate:SP[1-1]:TRACe[1-1]:DATA:UMEMory
- :CALCulate:SP[1-1]:TRACe[1-1]:LIMit:LOWer:LDATa
- :CALCulate:SP[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:DATA
- :CALCulate:SP[1-1]:TRACe[1-1]:LIMit:REPort[:DATA]
- :CALCulate:SP[1-1]:TRACe[1-1]:LIMit:UPPer:LDATa
- :CALCulate:SP[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:DATA

For frequency/power measurement

- :CALCulate:FP[1-1]:DATA:RDATa

- :CALCulate:FP[1-1]:DATA:TDATa
- :CALCulate:FP[1-1]:DATA:XDATa
- :CALCulate:FP[1-1]:TRACe[1-4]:DATA:FDATa
- :CALCulate:FP[1-1]:TRACe[1-4]:DATA:FMEMory
- :CALCulate:FP[1-1]:TRACe[1-4]:DATA:UDATa
- :CALCulate:FP[1-1]:TRACe[1-4]:DATA:UMEMory
- :CALCulate:FP[1-1]:TRACe[1-4]:LIMit:LOWer:LDATa
- :CALCulate:FP[1-1]:TRACe[1-4]:LIMit:LOWer:SEGMent:DATA
- :CALCulate:FP[1-1]:TRACe[1-4]:LIMit:REPort[:DATA]
- :CALCulate:FP[1-1]:TRACe[1-4]:LIMit:UPPer:LDATa
- :CALCulate:FP[1-1]:TRACe[1-4]:LIMit:UPPer:SEGMent:DATA

For transient measurement

- :CALCulate:TR[1-1]:NARRow:DATA:RDATa
- :CALCulate:TR[1-1]:NARRow:DATA:XDATa
- :CALCulate:TR[1-1]:TRACe[1-8]:DATA:FDATa
- :CALCulate:TR[1-1]:TRACe[1-8]:DATA:FMEMory
- :CALCulate:TR[1-1]:TRACe[1-8]:DATA:UDATa
- :CALCulate:TR[1-1]:TRACe[1-8]:DATA:UMEMory
- :CALCulate:TR[1-1]:TRACe[1-8]:LIMit:LOWer:LDATa
- :CALCulate:TR[1-1]:TRACe[1-8]:LIMit:LOWer:SEGMent:DATA
- :CALCulate:TR[1-1]:TRACe[1-8]:LIMit:REPort[:DATA]
- :CALCulate:TR[1-1]:TRACe[1-8]:LIMit:UPPer:LDATa
- :CALCulate:TR[1-1]:TRACe[1-8]:LIMit:UPPer:SEGMent:DATA
- :CALCulate:TR[1-1]:WIDE:DATA:RDATa
- :CALCulate:TR[1-1]:WIDE:DATA:XDATa

- :CALCulate:TR[1-1]:NARRow[1-2]:DATA:RDATa
- :CALCulate:TR[1-1]:NARRow[1-2]:DATA:XDATa

For AM noise measurement

- :CALCulate:AM[1-1]:DATA:CARRier
- :CALCulate:AM[1-1]:DATA:PDATa
- :CALCulate:AM[1-1]:DATA:RDATa
- :CALCulate:AM[1-1]:DATA:XDATa
- :CALCulate:AM[1-1]:TRACe[1-1]:DATA:FDATa
- :CALCulate:AM[1-1]:TRACe[1-1]:DATA:FMEMory
- :CALCulate:AM[1-1]:TRACe[1-1]:DATA:PDATa
- :CALCulate:AM[1-1]:TRACe[1-1]:DATA:PMEMory
- :CALCulate:AM[1-1]:TRACe[1-1]:DATA:SDATa
- :CALCulate:AM[1-1]:TRACe[1-1]:DATA:SMEMory
- :CALCulate:AM[1-1]:TRACe[1-1]:DATA:UDATa
- :CALCulate:AM[1-1]:TRACe[1-1]:DATA:UMEMory
- :CALCulate:AM[1-1]:TRACe[1-1]:LIMit:LOWer:LDATa
- :CALCulate:AM[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:DATA
- :CALCulate:AM[1-1]:TRACe[1-1]:LIMit:REPort[:DATA]
- :CALCulate:AM[1-1]:TRACe[1-1]:LIMit:UPPer:LDATa
- :CALCulate:AM[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:DATA
- :CALCulate:AM[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:DATA

For baseband noise measurement

- :CALCulate:BB[1-1]:DATA:PDATa
- :CALCulate:BB[1-1]:DATA:RDATa
- :CALCulate:BB[1-1]:DATA:XDATa
- :CALCulate:BB[1-1]:TRACe[1-1]:DATA:FDATa

- :CALCulate:BB[1-1]:TRACe[1-1]:DATA:FMEMory
- :CALCulate:BB[1-1]:TRACe[1-1]:DATA:PDATa
- :CALCulate:BB[1-1]:TRACe[1-1]:DATA:PMEMory
- :CALCulate:BB[1-1]:TRACe[1-1]:DATA:SDATa
- :CALCulate:BB[1-1]:TRACe[1-1]:DATA:SMEMory
- :CALCulate:BB[1-1]:TRACe[1-1]:DATA:UDATa
- :CALCulate:BB[1-1]:TRACe[1-1]:DATA:UMEMory
- :CALCulate:BB[1-1]:TRACe[1-1]:LIMit:LOWer:LDATa
- :CALCulate:BB[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:DATA
- :CALCulate:BB[1-1]:TRACe[1-1]:LIMit:REPort[:DATA]
- :CALCulate:BB[1-1]:TRACe[1-1]:LIMit:UPPer:LDATa
- :CALCulate:BB[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:DATA
- :CALCulate:BB[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:DATA

For user measurement

- :CALCulate:USER[1-1]:TRACe[1-8]:DATA:FDATa
- :CALCulate:USER[1-1]:TRACe[1-8]:DATA:FMEMory
- :CALCulate:USER[1-1]:TRACe[1-8]:DATA:RDATa
- :CALCulate:USER[1-1]:TRACe[1-8]:DATA:UDATa
- :CALCulate:USER[1-1]:TRACe[1-8]:DATA:UMEMory
- :CALCulate:USER[1-1]:TRACe[1-8]:DATA:XDATa
- :CALCulate:USER[1-1]:TRACe[1-8]:LIMit:LOWer:LDATa
- :CALCulate:USER[1-1]:TRACe[1-8]:LIMit:LOWer:SEGMent:DATA
- :CALCulate:USER[1-1]:TRACe[1-8]:LIMit:REPort[:DATA]
- :CALCulate:USER[1-1]:TRACe[1-8]:LIMit:UPPer:LDATa
- :CALCulate:USER[1-1]:TRACe[1-8]:LIMit:UPPer:SEGMent:DATA

For segment phase noise measurement

- :CALCulate:PS[1-1]:DATA:CARRier
- :CALCulate:PS[1-1]:DATA:PDATa
- :CALCulate:PS[1-1]:DATA:RDATa
- :CALCulate:PS[1-1]:DATA:XDATa
- :CALCulate:PS[1-1]:TRACe[1-1]:DATA:FDATa
- :CALCulate:PS[1-1]:TRACe[1-1]:DATA:FMEMory
- :CALCulate:PS[1-1]:TRACe[1-1]:DATA:PDATa
- :CALCulate:PS[1-1]:TRACe[1-1]:DATA:PMEMory
- :CALCulate:PS[1-1]:TRACe[1-1]:DATA:SDATa
- :CALCulate:PS[1-1]:TRACe[1-1]:DATA:SMEMory
- :CALCulate:PS[1-1]:TRACe[1-1]:DATA:UDATa
- :CALCulate:PS[1-1]:TRACe[1-1]:DATA:UMEMory
- :CALCulate:PS[1-1]:TRACe[1-1]:LIMit:LOWer:LDATa
- :CALCulate:PS[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:DATA
- :CALCulate:PS[1-1]:TRACe[1-1]:LIMit:REPort[:DATA]
- :CALCulate:PS[1-1]:TRACe[1-1]:LIMit:UPPer:LDATa
- :CALCulate:PS[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:DATA
- :CALCulate:PS[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:DATA

For sending and receiving data to/from VBA program

- :PROGram:VARiable:ARRay[1-10]:DATA
- :SENSe:CORRection:POWer:DATA

Block Type

If you use any of the following commands, the data transfer will be executed using block-type data. Here, you can choose either IEEE 64-bit floating point binary transfer format or IEEE 32-bit floating point binary transfer format.

- :MMEM:DATA

NOTE The instrument always uses the ASCII transfer format when you transfer data without using array- or block-type commands.

Data Transfer Format

- Overview
- ASCII Transfer Format
- Integer Format
- Floating-Point Number Format
- Binary Transfer Format

Other topics about Reading-Writing Measurement Data

Overview

When you transfer data using the one of the following commands, you can choose among ASCII transfer format, IEEE 64-bit floating point binary transfer format and IEEE 32-bit floating point binary transfer format.

Set the data transfer format to transfer data of either array or block type. To set the data transfer format, use the following command:

:FORMat:DATA

NOTE Executing the :SYSTem:PRESet or *RST does not affect the current setting of the data transfer format.

ASCII Transfer Format

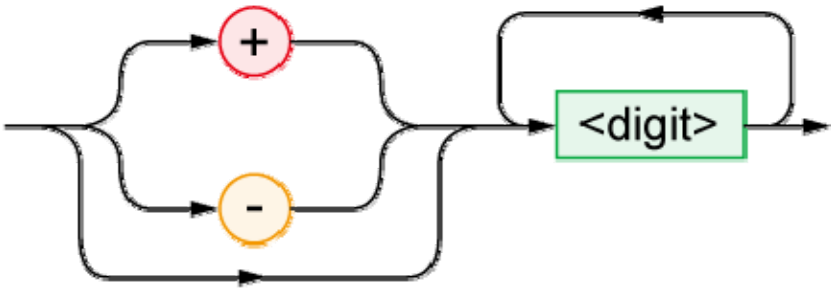
When you select the ASCII transfer format as the data transfer format, numbers are transferred as ASCII bytes, each of which corresponds to one of the formats shown below. Note that numbers are separated from one another with a comma (,) in accordance with the IEEE 488.2 specification.

NOTE Numeric data strings vary in length. Keep this in mind when you extract some data from retrieved numeric data strings in your program.

Integer Format

The figure below shows this format. Numbers are expressed as integers. For example, 201 is expressed as "+201" or "201."

Integer format

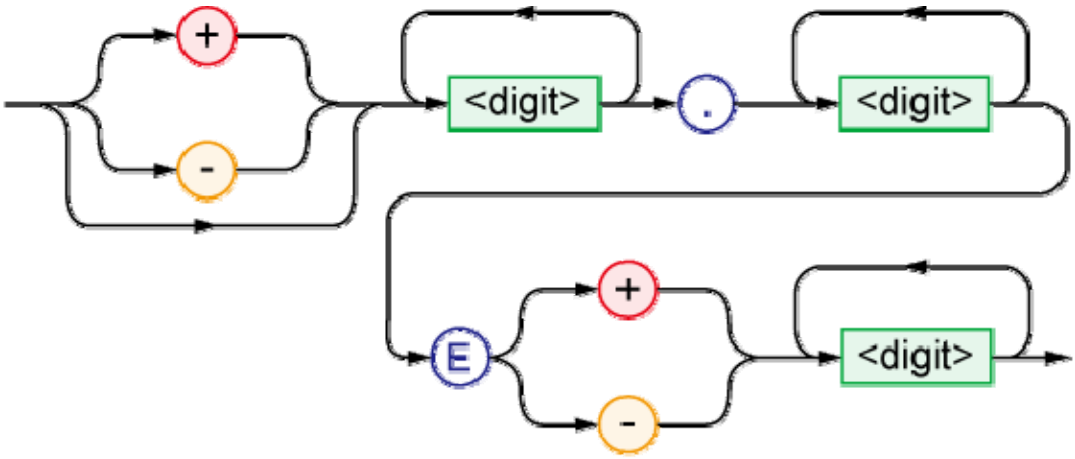


ssa0046

Floating-Point Number Format

The figure below shows this format. Numbers are expressed with floating points. For example, 1000 is expressed as "+1.00000000000E+003."

Floating-point number format



ssa0047

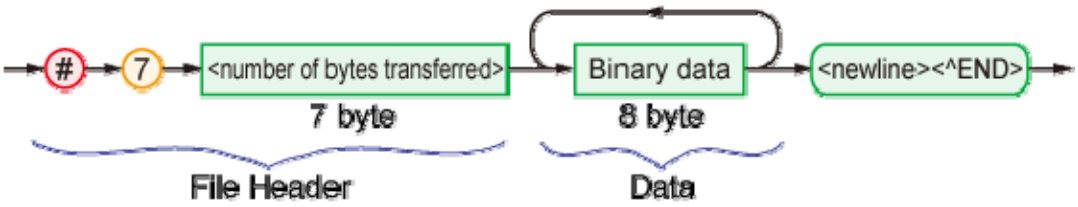
Binary Transfer Format

You can select the binary transfer format from the IEEE 64-bit floating point format or the IEEE 32-bit floating point format depending on the controller you use.

IEEE 64-bit floating point format

When you select the IEEE 64-bit floating point binary transfer format as the data transfer format, numbers are transferred in the format shown in the figure below.

Binary transfer format

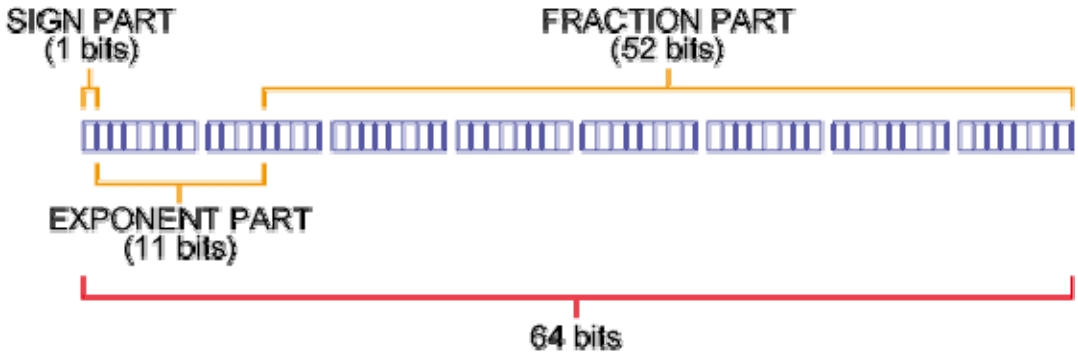


ssa0048

This data transfer format uses a header that consists of a sharp character (#), a number of 7 (which indicates the byte size of the <number of bytes transferred> part), and the <number of bytes transferred> part in this order. The header is followed by the binary data (each number consists of 8 bytes and the total is the byte size indicated by <number of bytes transferred>) and the message terminator <new line>^END.

The binary data is expressed in the IEEE 754 64-bit floating-point number format shown in the figure below.

64-bit floating point format

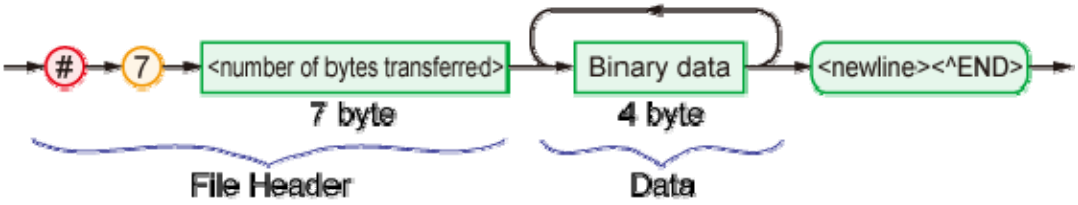


ssa0049

IEEE 32-bit floating point format

When you select the IEEE 32-bit floating point binary transfer format as the data transfer format, numbers are transferred in the format shown in the figure below.

IEEE 32-bit floating point binary transfer format

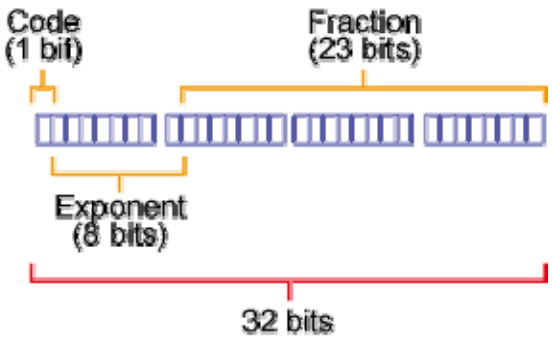


ssa0050

This data transfer format uses a header that consists of a sharp character (#), a number of 7 (which indicates the byte size of the <number of bytes transferred> part), and the <number of bytes transferred> part in this order. The header is followed by the binary data (each number consists of 4 bytes and the total is the byte size indicated by <number of bytes transferred>) and the message terminator <new line>^END.

The binary data is expressed in the IEEE 754 32-bit floating-point number format shown in the figure below.

32-bit floating point data



ssa0051

Byte order

When you opt to perform binary transfer, you can configure the instrument to transfer the bytes of the data in one of the following two byte orders:

NORMal

Transfer begins with the byte that contains the MSB (Most Significant Bit); that is, the leftmost byte in 64 bit floating point format and 32 bit floating point data.

SWAPped

Transfer begins with the byte that contains the LSB (Least Significant Bit); that is, the rightmost byte in 64 bit floating point format and 32 bit floating point data.

To set the byte order, use the following command:

:FORMat:BORDer

NOTE Executing the :SYSTem:PRESet or *RST does not affect the current setting of the byte order.

Internal Data Processing

- Overview
- Data Flow
- Internal Data Arrays

Other topics about Reading-Writing Measurement Data

Overview

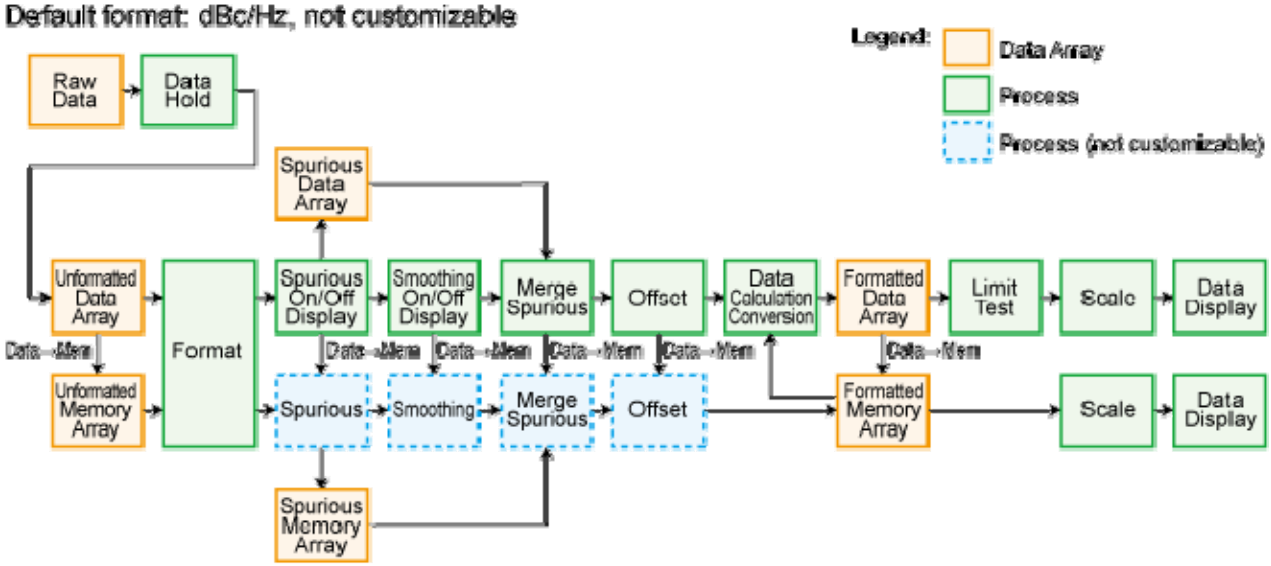
This section explains the internal data processing of the E5052B.

Data Flow

The following figures provide overviews of the E5052B's internal data processing flows.

Data processing flow for phase noise, AM noise, baseband noise and segment phase noise measurement with E5052B

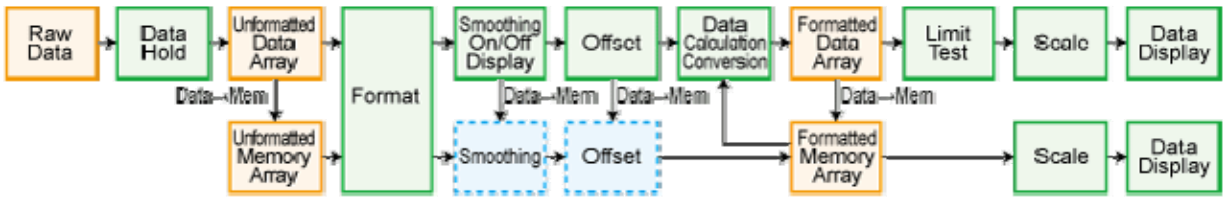
Default format: dBc/Hz, not customizable



ssa0052

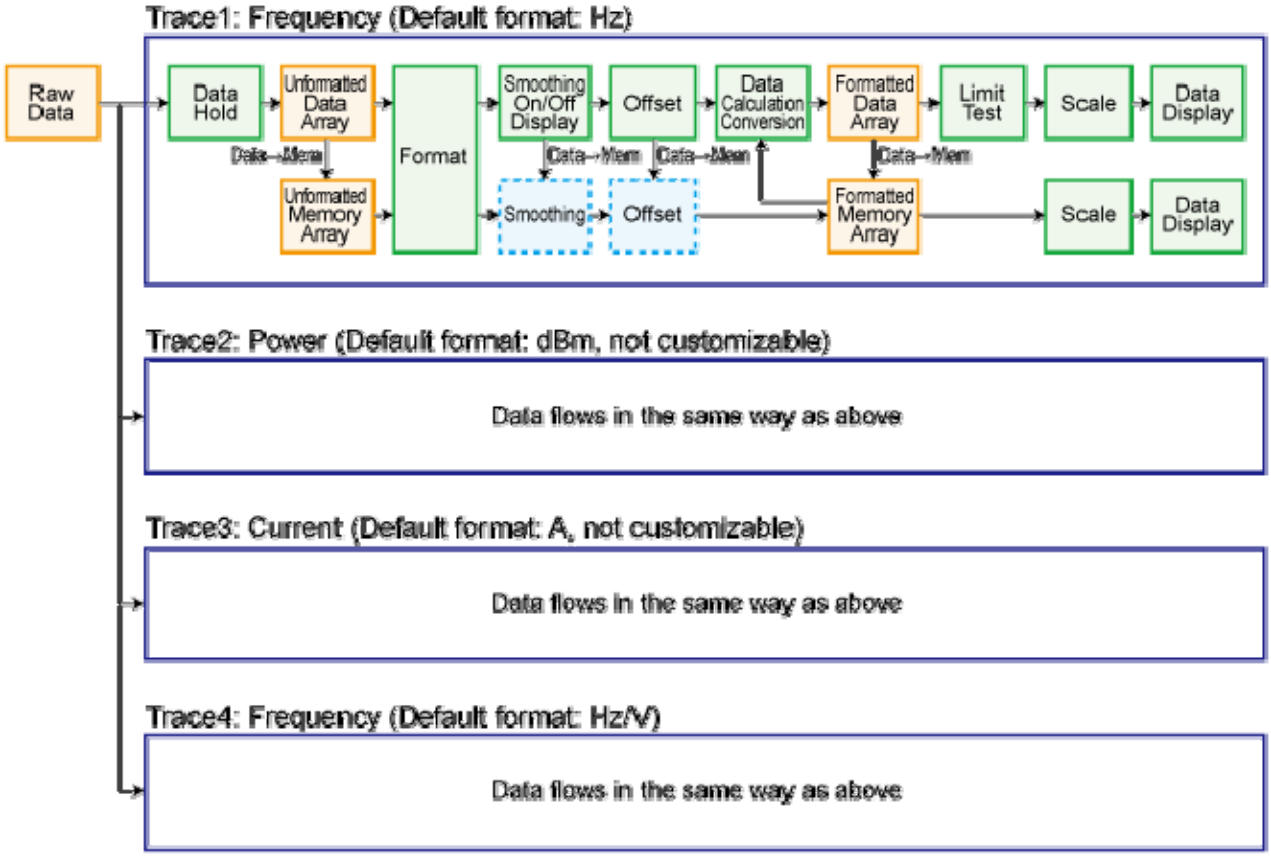
Data processing flow for spectrum measurement with E5052B

Default format: dBm



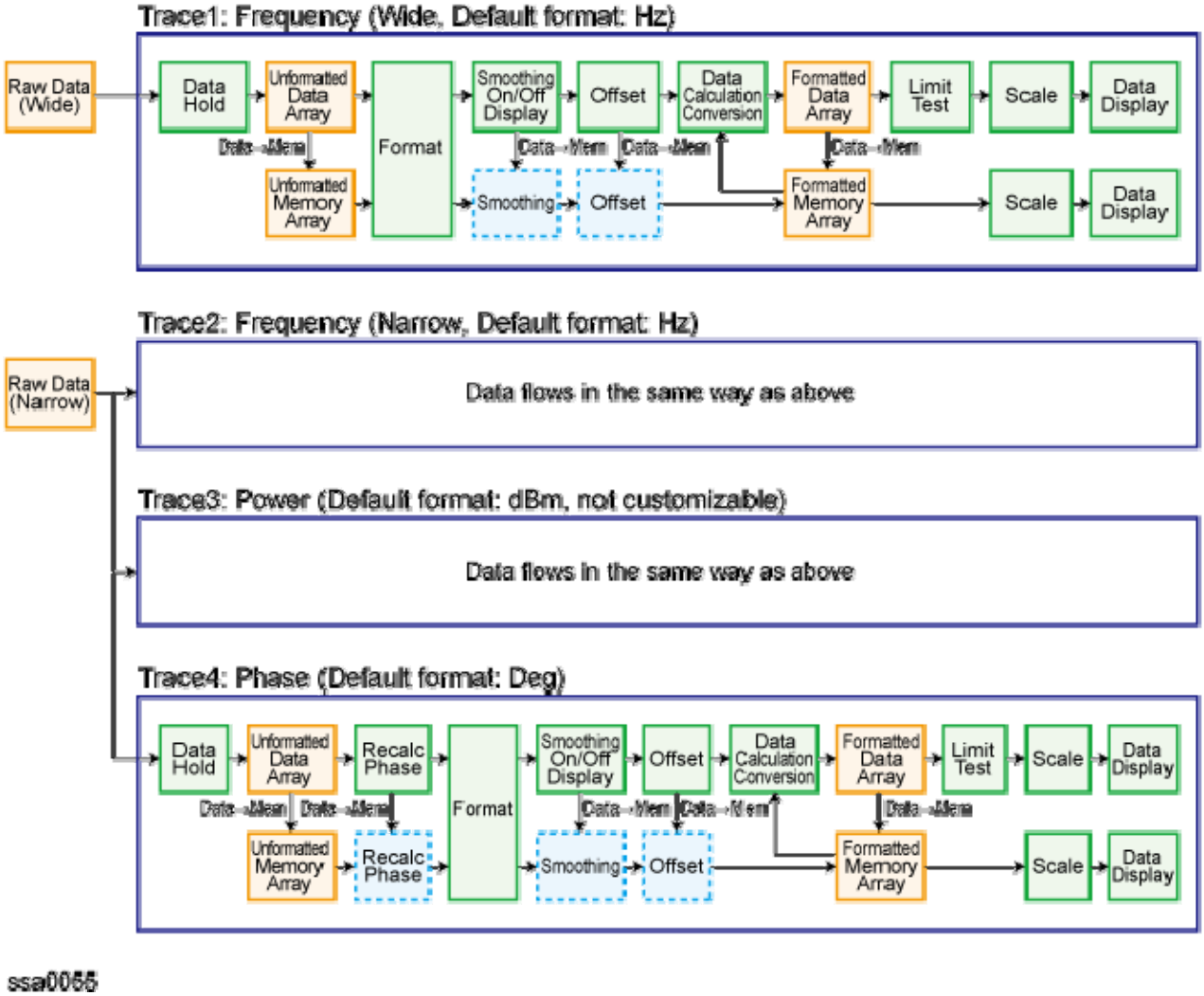
ssa0053

Data processing flow for frequency/power measurement with E5052B



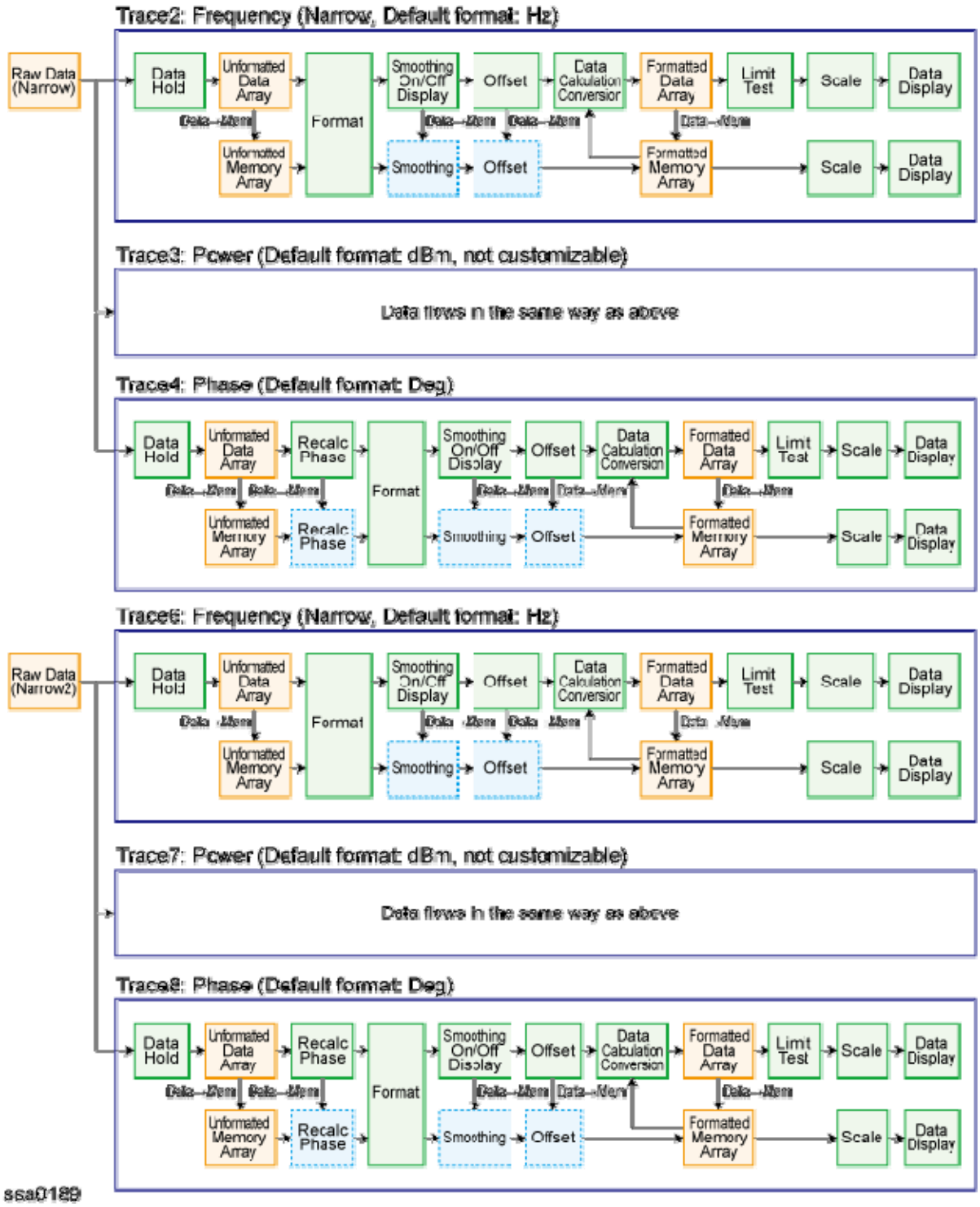
ssa0054

Data processing flow for transient measurement with E5052B (Wide-Narrow)

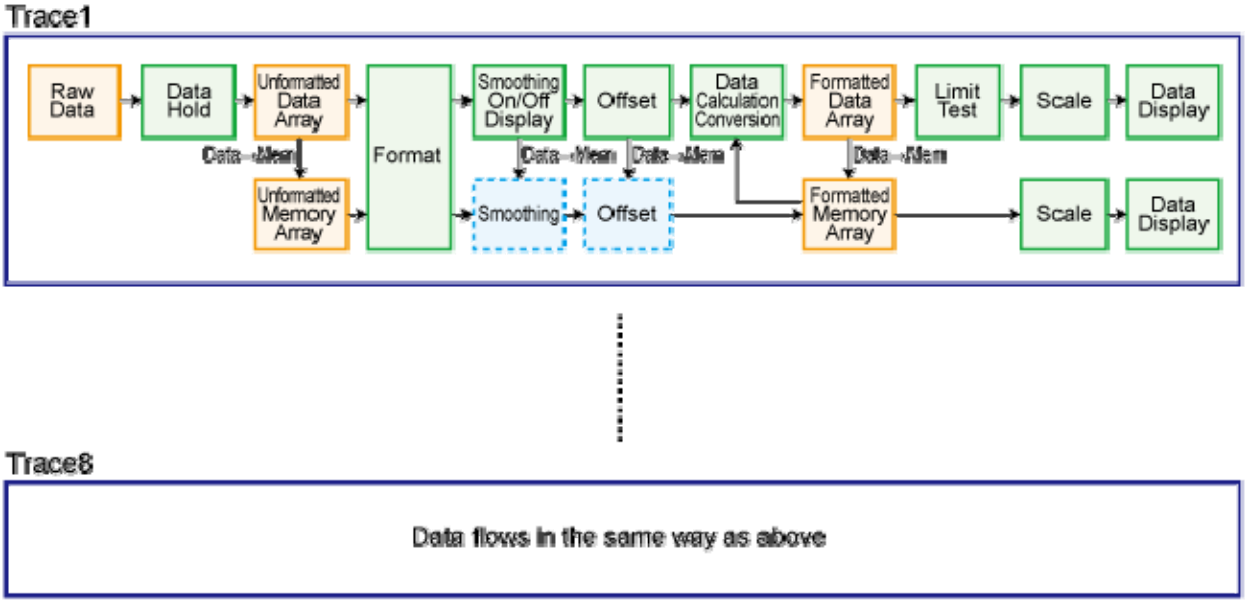


ssa0055

Data processing flow for transient measurement with E5052B (Narrow-Narrow)



Data processing flow for user-defined function measurement with E5052B



ssa0056

The enclosed part of the data processing flow represents the data array. By using SCPI commands, the data array can be read for processing and written for display on the E5052B's screen.

Internal Data Arrays

Unformatted data arrays

An unformatted array contains the raw data just as it was obtained through measurement and divided for each trace.

The instrument retains the unformatted data arrays which are associated with each trace of individual measurements. To read/write one of the unformatted data arrays, use the following command ("xx" specifies any one of the measurements (AM,BB,PN,PS,SP,FP,TR,USER) while "y" specifies the number of the trace if the measurement has multiple traces.

- :CALC:xx[1-1]:TRAC[1-y]:DATA:UDAT

  There is an unformatted data array formatted in dBc only when the AM noise, Baseband noise, Phase noise and Segment phase noise measurement are performed.

- :CALCulate:AM[1-1]:TRACe[1-1]:DATA:PDATa

- :CALCulate:BB[1-1]:TRACe[1-1]:DATA:PDATa
- :CALCulate:PN[1-1]:TRACe[1-1]:DATA:PDATa
- :CALCulate:PS[1-1]:TRACe[1-1]:DATA:PDATa

Unformatted memory arrays

When the :CALC:xx[1-1]:TRAC[1-y]:MATH:MEM command is executed on a particular unformatted memory array, a copy is stored in the corresponding unformatted data array.

The instrument retains the unformatted memory arrays associated with each trace of individual measurements. To read/write one of the unformatted memory arrays, use the following command:

- :CALC:xx[1-1]:TRAC[1-y]:DATA:UMEM

There is an unformatted memory array formatted in dBc only when the AM noise, Baseband noise, Phase noise and Segment phase noise measurement are performed.

- :CALCulate:AM[1-1]:TRACe[1-1]:DATA:PMEMory
- :CALCulate:BB[1-1]:TRACe[1-1]:DATA:PMEMory
- :CALCulate:PN[1-1]:TRACe[1-1]:DATA:PMEMory
- :CALCulate:PS[1-1]:TRACe[1-1]:DATA:PMEMory

Formatted data array

A formatted data array contains the formatted data (values to be displayed) obtained by performing data math operations, measurement parameter conversion, and smoothing on a particular unformatted data array.

The instrument retains the formatted data arrays associated with each trace of individual measurements. To read/write one of the formatted data arrays, use the following command:

- :CALC:xx[1-1]:TRAC[1-y]:DATA:FDAT

Formatted memory arrays

A formatted memory array contains the formatted data (values to be displayed) obtained by performing data math operations, measurement parameter conversion, and smoothing on a particular unformatted memory array.

The instrument retains the formatted memory arrays associated with each trace of individual measurements. To read/write one of the formatted memory arrays, use the following command:

- :CALC:xx[1-1]:TRAC[1-y]:DATA:FMEM

X-axis Data Arrays

An x-axis data array contains the x-axis values for all measurement points.

The instrument retains the X-axis arrays associated with individual measurements (for the transient measurement, narrow/wide only). X-axis data arrays are read-only. To retrieve one of the X-axis data arrays, use the following command:

- :CALC:xx[1-1]:DATA:XDAT

For the transient measurement, you can use two types of command as follows.

- :CALCulate:TR[1-1]:NARRow[1-2]:DATA:XDATa
- :CALCulate:TR[1-1]:WIDE:DATA:XDATa

Raw Data Arrays

A raw data array contains the measured values just as they were obtained, without any correction or format applied.

The instrument retains the raw data arrays associated with individual measurements (for the transient measurement, Wide/Narrow or Narrow/Narrow). To read/write one of the raw data arrays, use the following command:

- :CALC:xx[1-1]:DATA:RDAT

For the transient measurement, you can use two types of command as follows:

- :CALCulate:TR[1-1]:NARRow[1-2]:DATA:RDATa
- :CALCulate:TR[1-1]:WIDE:DATA:RDATa

There is a raw data array formatted in dBc only when the AM noise, Baseband noise, Phase noise and Segment phase noise measurement are performed.

- :CALCulate:AM[1-1]:DATA:PDATa
- :CALCulate:BB[1-1]:DATA:PDATa
- :CALCulate:PN[1-1]:DATA:PDATa
- :CALCulate:PS[1-1]:DATA:PDATa

Entering Data in a Trace

By using Formatted data array, you can change the data/memory trace on the LCD by writing the new data into the Formatted memory arrays.

When you write data into the formatted data/memory array, you can choose either the ASCII or binary transfer format (see Data Types for Data Transfer).

Using ASCII Transfer Format to Write Formatted Data Arrays and Using Binary Transfer Format to Write Formatted Data Arrays (write_b.htb) show sample programs that demonstrate how to write data into formatted data arrays. The sample program in Using ASCII Transfer Format to Write Formatted Data Arrays (write_a.htb) uses the ASCII transfer format, while the sample in Using *Binary* Transfer Format to Write Formatted Data Arrays (write_b.htb) uses the binary transfer format.

These sample programs write the trace data of phase noise measurements stored in a file on the formatted data arrays.

Sample: Using ASCII Transfer Format to Write Formatted Data Arrays (write_a.htb)

```
110 REAL Fdata(1:1601),Freq(1:1601)
120 DIM Img$[30],File$[300]
130 INTEGER Nop,I
140 File$="a:pn_asc"
150 ASSIGN @Agte5052 TO 717
160 OUTPUT @Agte5052;":SENS:PN:SWE:POIN?"
170 ENTER @Agte5052;Nop
180 PRINT "Number of Frequency points = ";Nop
190 REDIM Fdata(1:Nop),Freq(1:Nop)
200 ASSIGN @File TO File$
210 Img$="MD.4DE,2X,MD.6DE"
220 FOR I=1 TO Nop
230 ENTER @File USING Img$;Freq(I),Fdata(I)
240 NEXT I
250 ASSIGN @File TO *
260 OUTPUT @Agte5052;":FORM:DATA ASC"
270 OUTPUT @Agte5052;":CALC:PN1:TRAC1:DATA:FDAT ";Fdata(*)
```

```
390 END
```

Sample: Using Binary Transfer Format to Write Formatted Data Arrays (write_b.htb)

```
110 REAL Fdata(1:1601),Freq(1:1601)
120 DIM Img$[30],File$[300],Head$[10]
130 INTEGER Nop,I
140 File$="a:pn_asc"
150 ASSIGN @Agte5052 TO 717
160 ASSIGN @Binary TO 717;FORMAT OFF
170 OUTPUT @Agte5052;":SENS:PN:SWE:POIN?"
180 ENTER @Agte5052;Nop
190 PRINT "Number of Frequency points = ";Nop
200 REDIM Fdata(1:Nop),Freq(1:Nop)
210 ASSIGN @File TO File$
220 Img$="MD.4DE,2X,MD.6DE"
230 FOR I=1 TO Nop
240 ENTER @File USING Img$;Freq(I),Fdata(I)
250 NEXT I
260 ASSIGN @File TO *
270 OUTPUT @Agte5052;":FORM:DATA REAL64"
280 Head$="#6"&IVAL$(8*Nop,10)
290 OUTPUT @Agte5052;":CALC:PN1:TRAC1:DATA:FDAT ";Head$;
300 OUTPUT @Binary;Fdata(*),END
310 END
```

Other topics about Reading-Writing Measurement Data

Retrieving Measurement Results

- Overview
- Retrieving Measurement Results at Marker Positions
- Retrieving Internal Data Arrays

Other topics about Reading-Writing Measurement Data

Overview

Markers allow you to retrieve measurement results at your specified points only. Internal data arrays allow you to retrieve all measurement results throughout a particular trace.

Retrieving Measurement Results at Marker Positions

In using markers to retrieve measurement results at specified points, you can use up to ten markers for each measurement graph, and you can move them to any point on the trace. You can also set any as a reference marker.

Showing / hiding markers

To show or hide each marker, use the following command ("xx" specifies any one of the measurements (AM,BB,PN,PS,SP,FP,TR,USER) while "y" specifies the number of the trace if the measurement has multiple traces.

- :CALC:xx[1-1]:TRAC[1-y]:MARK[1-10]:STAT

Turning On or Off Reference Marker Mode

Turning on Reference *Marker* Mode sets the specified marker number as the reference marker and makes other markers relative with respect to it. To specify the reference marker and turn on or off Reference Marker mode, use the following command:

- :CALC:xx[1-1]:ALLT:MARK:REF:NUMB
- :CALC:xx[1-1]:ALLT:MARK:REF:STAT

Changing along the frequency axis and retrieval of marker positions

To change along the frequency axis or retrieve the reference marker position, use the following command:

- :CALC:xx[1-1]:TRAC[1-y]:MARK[1-10]:X

NOTE When Reference Marker Mode is on, the x value at a regular marker is a relative value obtained by deducting its x value from the reference marker's x value.

Retrieving measurement results at marker positions

To retrieve the measurement results (response values) at a particular marker or the reference marker, use the following command:

- :CALC:xx[1-1]:TRAC[1-y]:MARK[1-10]:Y?

Retrieving Internal Data Arrays

You can chose between the ASCII and binary data transfer formats when you retrieve internal data arrays. For more information, refer to Data Types for Data Transfer.

Using ASCII Transfer Format to Retrieve Internal Data Arrays and Using Binary Transfer Format to Retrieve Internal Data Arrays show sample programs that demonstrate how to retrieve formatted data arrays. The sample program in Using ASCII Transfer Format to Retrieve Internal Data Arrays uses the ASCII transfer format while the sample in Using Binary Transfer Format to Retrieve Internal Data Arrays uses the binary transfer format.

These sample programs show how to retrieve the formatted data arrays of measured phase noise.

Using ASCII Transfer Format to Retrieve Internal Data Arrays

```
110 REAL Fdata(1:1601),Freq(1:1601)
120 DIM Img$[30]
130 INTEGER Nop,I
140 ASSIGN @Agte5052 TO 717
150 OUTPUT @Agte5052;":SENS:PN:SWE:POIN?"
160 ENTER @Agte5052;Nop
170 PRINT "Number of Frequency points = ";Nop
180 REDIM Fdata(1:Nop),Freq(1:Nop)
190 OUTPUT @Agte5052;":FORM:DATA ASC"
200 OUTPUT @Agte5052;":CALC:PN1:TRAC1:DATA:FDAT?"
210 ENTER @Agte5052;Fdata(*)
220 OUTPUT @Agte5052;":CALC:PN1:DATA:XDAT?"
230 ENTER @Agte5052;Freq(*)
240 Img$="MD.4DE,2X,MD.6DE"
```

```
250 PRINT "Frequency Data"
260 FOR I=1 TO Nop
270 PRINT USING Img$;Freq(I),Fdata(I)
280 NEXT I
290 END
```

Using Binary Transfer Format to Retrieve Internal Data Arrays

```
110 REAL Fdata(1:1601),Freq(1:1601)
120 DIM Img$[30],Buff$[9]
130 INTEGER Nop,I
140 ASSIGN @Agte5052 TO 717
150 ASSIGN @Binary TO 717;FORMAT OFF
160 OUTPUT @Agte5052;":SENS:PN:SWE:POIN?"
170 ENTER @Agte5052;Nop
180 PRINT "Number of Frequency points = ";Nop
190 REDIM Fdata(1:Nop),Freq(1:Nop)
200 OUTPUT @Agte5052;":FORM:DATA REAL64"
210 OUTPUT @Agte5052;":CALC:PN1:TRAC1:DATA:FDAT?"
220 ENTER @Agte5052 USING "#,8A";Buff$
230 ENTER @Binary;Fdata(*)
240 ENTER @Agte5052 USING "#,1A";Buff$
250 OUTPUT @Agte5052;":CALC:PN1:DATA:XDAT?"
260 ENTER @Agte5052 USING "#,8A";Buff$
270 ENTER @Binary;Freq(*)
280 ENTER @Agte5052 USING "#,1A";Buff$
290 Img$="MD.4DE,2X,MD.6DE"
300 PRINT "Frequency Data"
310 FOR I=1 TO Nop
320 PRINT USING Img$;Freq(I),Fdata(I)
330 NEXT I
340 END
```

Saving and Recalling

- Saving and Recalling File
- Managing Files

Saving and Recalling File

- Specifying File
- Saving and Recalling Instrument Status
- Saving Measurement Data
- Saving Images
- Saving/Loading (Importing) a VBA Program

Other topics about Saving and Recalling

Specifying File

When running a command for saving, recalling, and managing files, use a filename with extension to specify a particular file. Specify "F:" in the beginning of the file name, when specifying a file on the user area of hard disk. Also, when specifying a file name with directory, use "/" (slash) or "\" (backslash) as a delimiter.

Saving and Recalling Instrument Status

You can save the instrument state using one of the following 2 methods:

- Saving the entire instrument state into a file
- Saving only the instrument setting into the file

Selecting content to be saved

When saving the instrument status into a file, the content to be saved can be selected among the following 2 options:

- Entire instrument state (setting and data)
- Instrument setting only.

To select a content to be saved, use the following command:

:MMEMory:STORe:STYPe

Saving and recalling entire instrument status

To save the instrument setting or instrument setting and data ,use the following command:

:MMEMory:STORe:STATe

Recalling a file saved with the above command can reproduce the status when it was saved. To recall the settings from a file, use the following command:

:MMEMory:LOAD:STATe

Auto recall

The file saved with the name F:\autorec.sta or A:\autorec.sta will be automatically recalled when the E5052B is powered ON.

Saving Measurement Data

Measurement data (in a formatted data array) can be saved to a file in CSV (Comma Separated Value) format.

To save measurement data in a file, use the following command:

:MMEMory:xx[1-1]:TRACe[1-y]:STORe

Here, xx indicates anyone of the measurement PN,PS,SP,FP,TR,AM,BB,USER and y indicates the number of traces.

Executing the above command will save the measurement data of the active trace.

NOTE The data saved using the above command cannot be recalled from the E5052B.

Saving Images

Images displayed on the LCD screen can be saved to a file in the bitmap (.bmp) or portable network graphics (.png) format.

To save the screen image to a file, use the following command:

:MMEMory:STORe:IMAGe

Executing the above command will save the screen image when the command is invoked.

NOTE This gives different screen image results from those obtained by pressing the **Capture/System** key on the front panel.

Saving/Loading (Importing) a VBA Program

Saving

Only the VBA project file can be saved using command.

To save the VBA project that is opened on the VBA editor on the file, use the following command.

:MMEMory:STORe:PROGram

Loading (importing)

To load the VBA project to the VBA editor, or to import the module/form file, use the following command.

:MMEMory:LOAD:CORRection:POWer

Executing above command will load/import the file according to its extension as follows:

Extension	File type
vba	VBA Project
bas	Standard module
frm	User Forms
cls	Class Modules

Managing Files

- Various Commands
- Sample Program

Other topics about Saving and Recalling

Various Commands

Creating directory (folder)

To create a directory (folder), use the following command:

:MMEMory:MDIRectory

Deleting file (directory)

To delete a file or a directory, use the following command:

:MMEMory:DELete

Copying file

To copy a file, use the following command:

:MMEMory:COPY

Transferring files

File transfer from the external controller to the E5052B can be possible by reading data from a file on the controller and then writing them to the file on the E5052B.

:MMEMory:DATA

Also, file transfer from the E5052B to the external controller can be possible by reading data from a file on the E5052B using the commands as query and then writing them to the file on the controller.

Retrieving data from storage

To retrieve information for the storage that is built in the E5052B (usage, property of file located in a specified directory), use the following command;

:MMEMory:CATAlog

Sample Program

The following example shows a sample program for transferring files between the external controller and the E5052B.This program reads out data from a specified file on the E5052B and then writes them to a specified file on the external controller.

Sample of file transfer

```
1000 DIM Src_file$[50],Dst_file$[50]
1010 DIM Buff$[9],Img$[32],Src_size_char$[10]
1020 INTEGER Max_bsize,Block_size
1030 REAL Src_size
1040 ASSIGN @Agte5052 TO 717
1050 Src_file$="f:\state01.sta"
1060 Dst_file$="a:state01.sta"
1070 CREATE Dst_file$,1
1080 ASSIGN @Dst_file TO Dst_file$
1090 Max_bsize=24576
```

```
1100 PRINT "Now Copying: "&Src_file$&"(@E5052) ->
"&Dst_file$&"(@Controller)"
1110 OUTPUT @Agte5052;":MMEM:DATA? """&Src_file$&""""
1120 WAIT .5
1130 ENTER @Agte5052 USING "#,A";Buff$
1140 ENTER @Agte5052 USING "#,A";Digit$
1150 Img$="#, "&Digit$&"A"
1160 ENTER @Agte5052 USING Img$;Src_size_char$
1170 Src_size=VAL(Src_size_char$)
1180 WHILE Src_size>0
1190 IF Src_size>Max_bsize THEN
1200 Block_size=Max_bsize
1210 ELSE
1220 Block_size=Src_size
1230 END IF
1240 ALLOCATE Dat$[Block_size]
1250 Img$="#, "&VAL$(Block_size)&"A"
1260 ENTER @Agte5052 USING Img$;Dat$
1270 OUTPUT @Dst_file USING Img$;Dat$
1280 DEALLOCATE Dat$
1290 Src_size=Src_size-Block_size
1300 END WHILE
1310 PRINT "Done"
1320 ENTER @Agte5052 USING "#,A";Buff$
1330 ASSIGN @Dst_file TO *
```

Communication with External Devices (24 bit I/O)

- 24 bit I/O Port Overview
- I/O Signal Pin Layout and Description
- Inputting/Outputting Data
- Preset states at power-on
- Timing Chart
- Electrical Characteristics

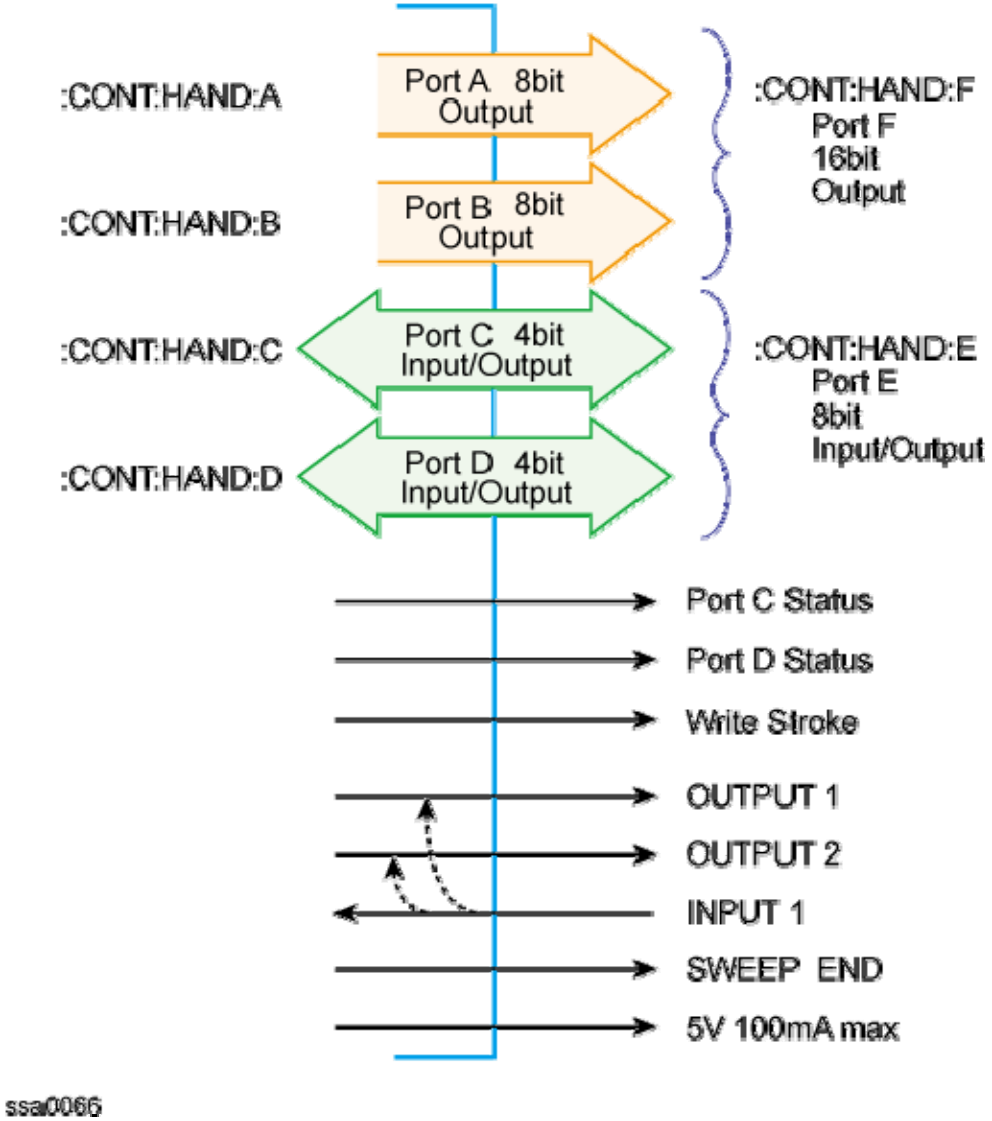
24 bit I/O Port Overview

The E5052B 24-bit *I/O* port provides four independent parallel ports for data I/O associated with several control signal lines and the power line. All signals operate in TTL logic.

The data I/O ports are configured with 2 pairs of 8-bit output ports and 2 pairs of 4-bit bi-directional ports. Furthermore, these ports can cooperate to provide a maximum 16-bit-wide output port or a maximum 8-bit-wide input port.

The I/O signals operate on a negative logic basis. The control signal lines consist of various control output data, including completion of measurement or control signal for handshaking.

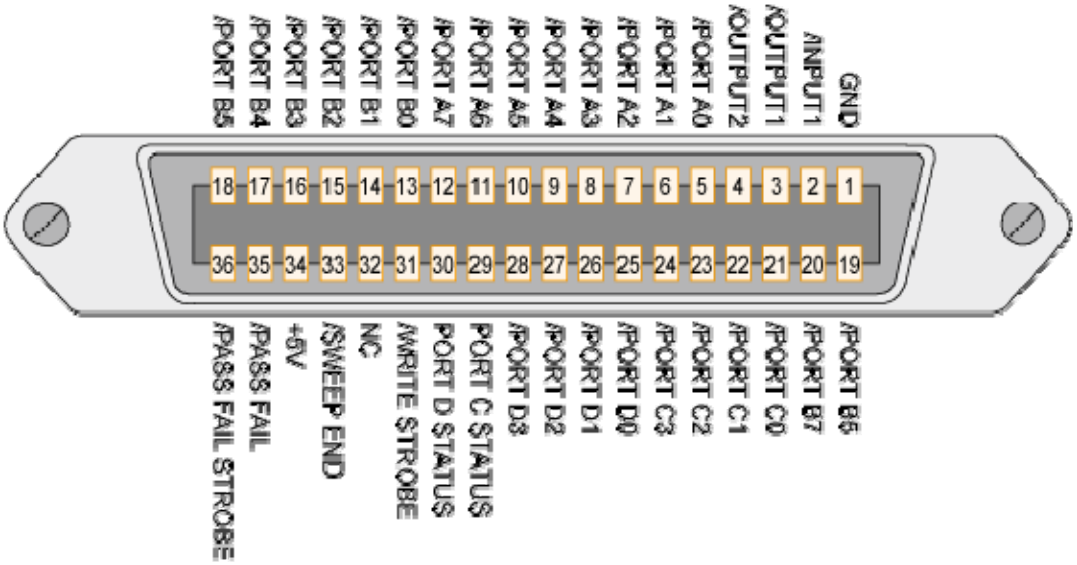
I/O ports and control signal lines



Other topics about Communication with External Devices

I/O Signal Pin Layout and Description

The layout of the *I/O* signal pins on the 24 bit interface connector and its description are shown below.



ssa0067

A slash (/) symbol preceding signal names means that they are negative logic (active low).

Pin number	Signal name	Input/Output	Description
1	GND	N/A	Ground.
2	/INPUT1	Input	When this port receives a negative pulse, /OUTPUT1 and /OUTPUT2 are changed to the Low level.
3	/OUTPUT1	Output	Changes to the Low level when /INPUT1 receives a negative pulse. A command can be available for altering the Low/High level logic.

4	/OUTPUT2	Output	Changes to the Low level when /INPUT1 receives a negative pulse. A command can be available for altering the Low/High level logic.
5	/PORT A0	Output	Bit 0 of port A (8 bit parallel output port)
6	/PORT A1	Output	Bit 1 of port A.
7	/PORT A2	Output	Bit 2 of port A.
8	/PORT A3	Output	Bit 3 of port A.
9	/PORT A4	Output	Bit 4 of port A.
10	/PORT A5	Output	Bit 5 of port A.
11	/PORT A6	Output	Bit 6 of port A.
12	/PORT A7	Output	Bit 7 of port A.
13	/PORT B0	Output	Bit 0 of port B (8 bit parallel output port)
14	/PORT B1	Output	Bit 1 of port B.
15	/PORT B2	Output	Bit 2 of port B.
16	/PORT B3	Output	Bit 3 of port B.
17	/PORT B4	Output	Bit 4 of port B.
18	/PORT B5	Output	Bit 5 of port B.
19	/PORT B6	Output	Bit 6 of port B.
20	/PORT B7	Output	Bit 7 of port B.
21	/PORT C0	Input/Output	Bit 0 of port C (4 bit parallel I/O port)
22	/PORT C1	Input/Output	Bit 1 of port C .

23	/PORT C2	Input/Output	Bit 2 of port C.
24	/PORT C3	Input/Output	Bit 3 of port C.
25	/PORT D0	Input/Output	Bit 0 of port D (4 bit parallel I/O port)
26	/PORT D1	Input/Output	Bit 1 of port D
27	/PORT D2	Input/Output	Bit 2 of port D.
28	/PORT D3	Input/Output	Bit 3 of port D.
29	PORT C STATUS	Output	Port C status signal. This signal is changed to the High level when port C is configured to output port. It is changed to the Low level when the port is configured to input port.
30	PORT D STATUS	Output	Port D status signal. This signal is changed to the High level when port D is configured to output port. It is changed to the Low level when the port is configured to input port.
31	/WRITE STROBE	Output	A output port write strobe signal. When data is present (that is, output level changes) on any of the output ports, this signal provides a negative pulse.
32	NC		Not Used
33	/SWEEP END	Output	A sweep completion signal. When measurement (all sweeps of all channels) and data calculation are completed, this signal provides a negative pulse.
34	+5V	Output	Provides +5V DC power supply

			for external instruments.
35	/PASS FAIL	Output	Each limit test's results signal. This signal changes to the High level when limit test, bandwidth test, or ripple test results return FAIL. It changes to the Low level when all limit test results return PASS.
36	/PASS FAIL STROBE	Output	Limit test result write strobe signal. When limit test result is present on /PASS FAIL, this signal provides a negative pulse.

Other topics about Communication with External Devices

Inputting/Outputting Data

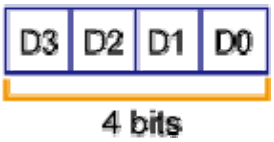
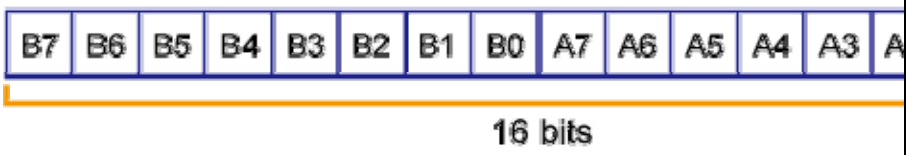
- Overview
- Specifying Signal Direction of Port
- Reading Data Input from Port
- Data Output to Port

Other topics about Communication with External Devices

Overview

The E5052B 24 bit *I/O* port provides the ports for data I/O shown below.

Port Name	Usage	Data Structure
Port A	Output	A7 A6 A5 A4 A3 A2 A1 A0 8 bits
Port B	Output	B7 B6 B5 B4 B3 B2 B1 B0 8 bits
Port C	Input/Output	C3 C2 C1 C0 4 bits

Port D	Input/Output	
Port E	Input/Output	
Port F	Output	

Specifying Signal Direction of Port

Signal direction (input/output) can be changed for the ports C, D, and D as shown in I/O ports and control signal lines. Thus, before the ports are used, the directions should be determined according to their usage.

To specify the signal direction for the ports C and D, use the following command. Direction for the port E depends on the setting for the ports C and D.

Port Name	*Command*
Port C	:CONTrol:HANDler:C:MODE
Port D	:CONTrol:HANDler:D:MODE

Reading Data Input into Port

When the ports C, D, or E is configured to input ports, binary data represented with High(0)/Low(1) of each bit of the port will be read as decimal data.

To retrieve the data, use the following command as query:

Port Name	Command
Port C	:CONTrol:HANDler:C[:DATA]
Port D	:CONTrol:HANDler:D[:DATA]
Port E	:CONTrol:HANDler:E[:DATA]

Data Output to Port

To the ports A through F (the ports C, D, and E should be configured to output ports), binary data (decimal data when output data is specified with a command) represented with High(0)/Low(1) of each bit of the port can be output.

To output data, use the following command:

Port Name	Command
Port A	:CONTrol:HANDler:A[:DATA]
Port B	:CONTrol:HANDler:B[:DATA]
Port C	:CONTrol:HANDler:C[:DATA]
Port D	:CONTrol:HANDler:D[:DATA]
Port E	:CONTrol:HANDler:E[:DATA]
Port F	:CONTrol:HANDler:F[:DATA]

Preset states at power-on

The 24 bit *I/O* port is set at power-on as follows (not affected at reset)

Description	Status
Port A	High (All Bits)
Port B	High (All Bits)
Port C	Input
Port D	Input
Port C STATUS	Low
Port D STATUS	Low
/OUTPUT1	High
/OUTPUT2	High
/SWEEP END	High
/PASS FAIL	High

Other topics about Communication with External Devices

Timing Chart

- Overview
- Timing Chart of Pulse Width of /SWEEP END
- Timing Chart of Data Output and Write Strobe Signal
- Timing Chart of /INPUT1 and /OUTPUT1 /OUTPUT2
- Timing Chart of Data Output and Write Strobe Signal (/PASS FAIL)

Other topics about Communication with External Devices

Overview

This section shows the typical timing chart of *I/O* port Signal.

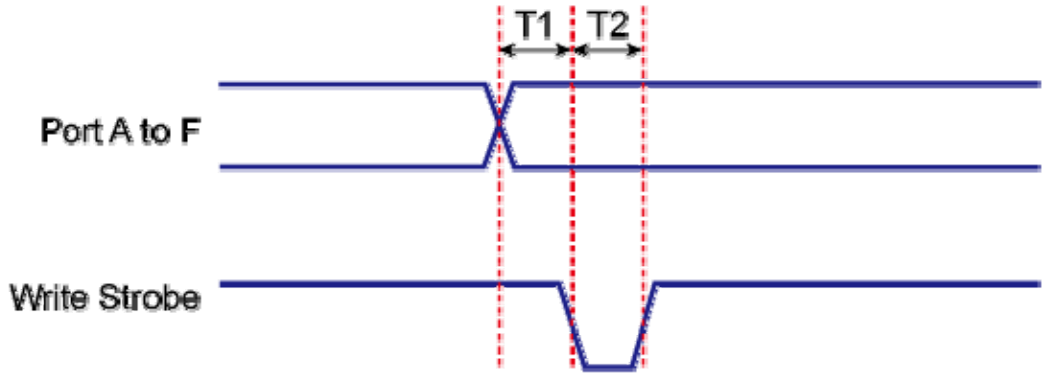
Timing Chart of Pulse Width of /SWEEP END

When the formatted data calculation is completed during the sweep, a negative pulse is provided. The pulse width of the sweep completion signal is shown in the figure below:

		Typical value
T1	Pulse width of /SWEEP END	12 µs

Timing Chart of Data Output and Write Strobe Signal

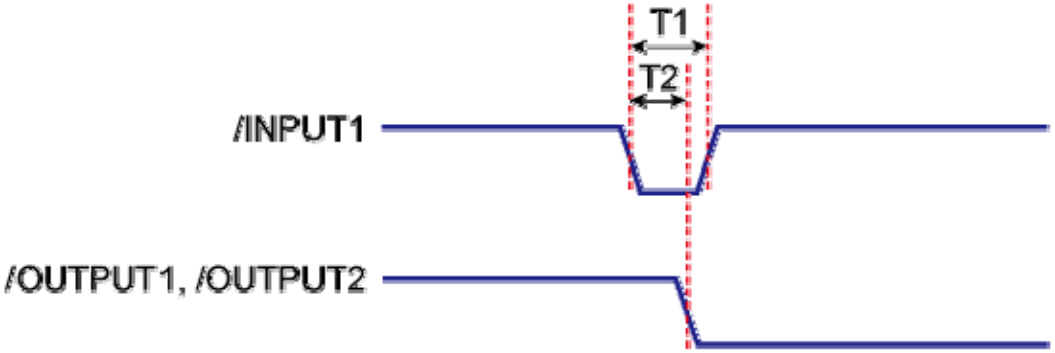
The figure below shows the timing chart for data output and write strobe signal output to ports A through F.



ssa0069

T1	Response time of write strobe signal	1 µs
T2	Pulse width of write strobe signal	1 µs

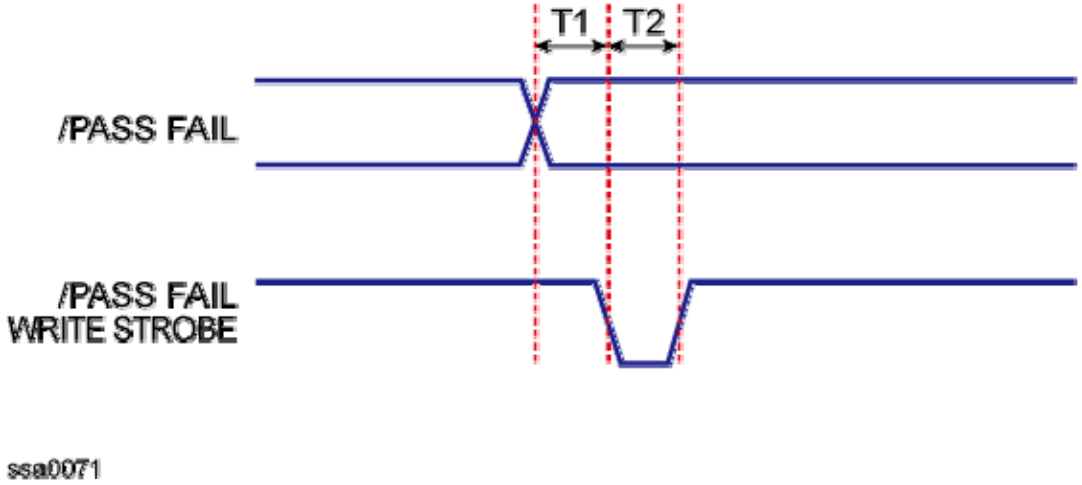
Timing Chart of /INPUT1 and /OUTPUT1, /OUTPUT2



ssa0070

		Minimum value
T1	Pulse width of /INPUT1	1 µs
T2	Response time of /OUTPUT1, /OUTPUT2	0.5 µs

Timing Chart of Data Output and Write Strobe Signal (/PASS FAIL)



T1	Response time of /PASS FAIL write strobe signal	1 µs
T2	Pulse width of /PASS FAIL write strobe signal	1 µs

Electrical Characteristics

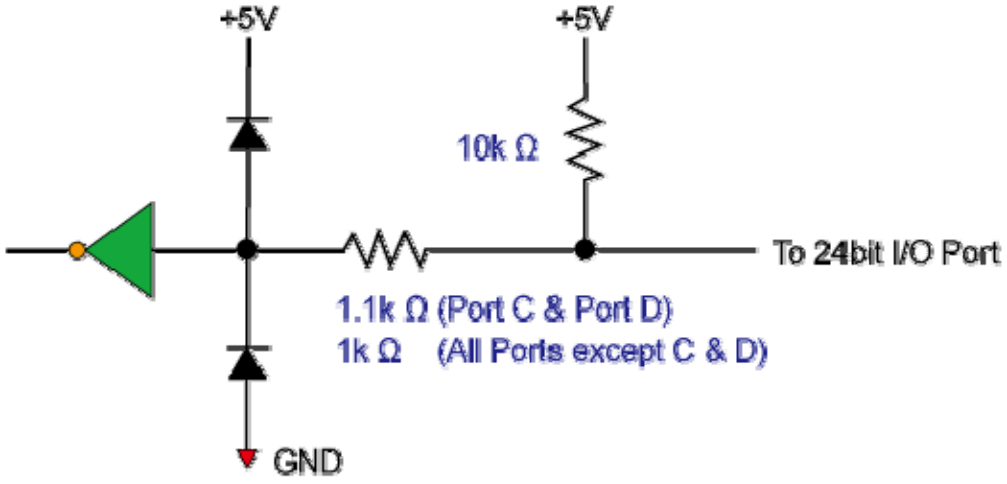
- Input Signal
- Output Signal
- Power Supply (+5 V)

Other topics about Communication with External Devices

Input Signal

All input signals are TTL compatible.

Maximum rate input voltage		-0.5 V to 5.5 V
Input voltage	**High level**	2.0 V to 5.0 V
	Low level	0 V to 0.5 V



ssa0072

Output Signal

All output signals are TTL compatible.

Maximum rate output current		-10 mA to 10 mA
Output current	**High level**	-5 mA
	Low level	-5 mA
Output voltage	**High level**	2.0 V to 3.3 V (when output current is from -5 mA to 0 mA) 3.20 V (when output current is -1 mA) 2.75 V (when output current is -5 mA)
	Low level	0 V to 0.8 V (when output current is from 0 mA to 3 mA) 0.25 V (when output current is 1 mA) 0.55 V (when output current is 3 mA)

Power Supply (+5 V)

The following table shows electrical characteristics of +5 V power supply for external instruments.

Output voltage	4.5 V to 5.5 V
Maximum output current	100 mA

Status Reporting System

- General Status Register Model
- Using the Status Reporting System
- Status Register Structure
- Status Bit Definition

General Status Register Model

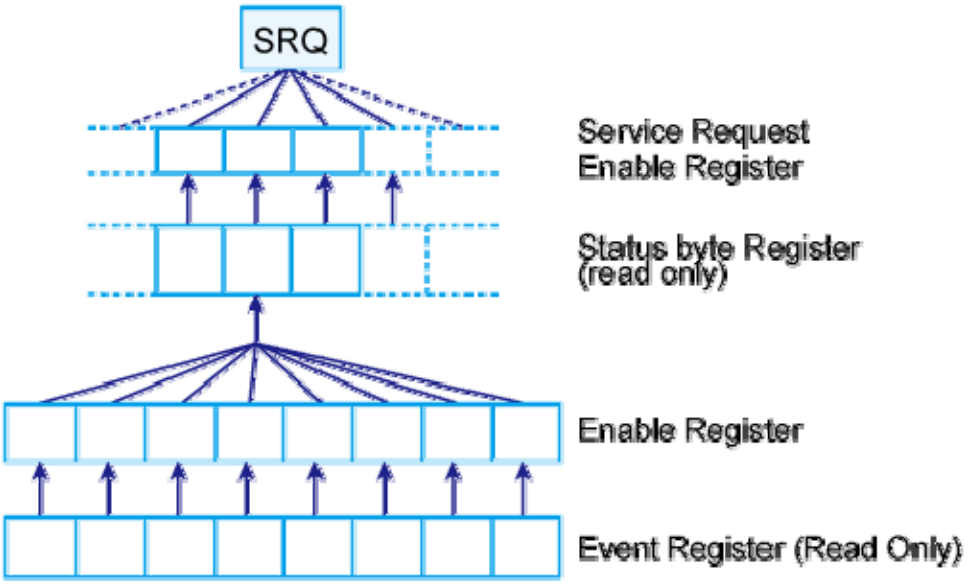
- Overview
- Event Register
- Enable Register
- Status Byte Register
- Condition Register and Transition Filter

Other topics about Status Reporting System

Overview

The Agilent E5052B has a status reporting system to report the condition of the instrument.

General status register model



ssa0058

The status reporting system has a hierarchical structure as shown in the figure above. When the instrument satisfies a particular condition, the corresponding bit of the event register is set to 1. Therefore, you can check the instrument status by reading the event register.

When the event register bit is set to "1" and a corresponding enable register bit (a bit marked with an arrow in General status register model) is also "1," the summary bit of the status byte register is set to "1." You can read the status byte register by using the serial poll.

If the bit of the service request enable register is "1," a service request (SRQ) is generated by the positive transition of the corresponding status byte register bit. By generating SRQ, you can notify the controller that the E5052B is requesting service. In other words, interruption by SRQ can be programmed. For more information on using SRQ, see Using the status register or Using the status reporting system.

NOTE A different status-reporting system is used for each parser, such as *GPIB* , VBA, Telnet, or SICL-LAN. For example, if you mask the status register with VBA, the masking is valid only when VBA is used.

Event Register

Reflects the corresponding condition of the E5052B (e.g., occurrence of an event) as a bit status. These bits continuously monitor changes in the E5052B's state and change the bit status when the condition (e.g., change bit status to "1" if a specific event occurs) for each bit is met. You cannot change the bit status by issuing a SCPI command.

Enable Register

Setting the enable register allows you to specify event register bits that can set "1" to the summary bit of the status byte register when an event occurs. The register bits work as mask bits; setting "1" to an enable register will enable a corresponding bit in the event register.

For example, when you want to set "1" as the summary bit in the status byte register by a specific register condition, set the corresponding enable register to "1."

Status Byte Register

If the enabled event register is set to "1," a corresponding bit of the status byte register is also set to "1." This register also indicates the output queue and SRQ status.

The value of the status byte register can be read by using the *STB command or serial poll (SPOLL statement in HTBasic) from the controller.

Reading the status byte register by using the *STB command does not affect the contents of the status byte register. However, reading it with the SPOLL statement of HTBasic will clear the RQS bit in the status byte register.

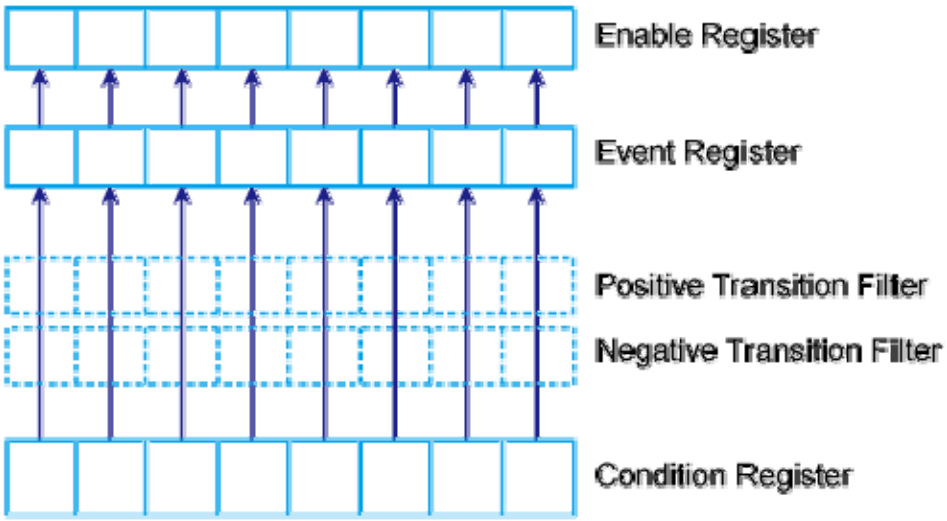
Also, setting the service request enable register using the *SRE command can generate a service request synchronously with the status byte register.

Condition Register and Transition Filter

When the status register has a transition filter, there is a lower register called a condition register under the event register. The transition filter is between the event register and the condition register.

The transition filter enables you to select a positive and/or negative transition of the condition register bit in order to set a bit in the corresponding event register. For example, using the negative transition filter to set bit 3 to "1" causes bit 3 of the event register to be set to "1" when bit 3 of the condition register makes a negative transition, that is, changes from 1 to 0.

Transition filter and condition register



ssa0059

In the E5052B, the following registers provide a condition register and transition filter:

- Operation status register
- Operation user defined status register
- Questionable status register

Using the Status Reporting System

You can manage the status reporting system using the following commands in any combination:

- *CLS
- *SRE
- *STB
- *ESE
- *ESR
- :STATus:OPERation:BIT12:CLEar
- :STATus:OPERation:BIT12:CONDition
- :STATus:OPERation:BIT12:ENABle
- :STATus:OPERation:BIT12[:EVENt]
- :STATus:OPERation:BIT12:NTRansition
- :STATus:OPERation:BIT12:PTRansition
- :STATus:OPERation:BIT12:SET
- :STATus:OPERation:CONDition
- :STATus:OPERation:ENABle
- :STATus:OPERation [:EVENt]
- :STATus:OPERation:NTRansition
- :STATus:OPERation:PTRansition
- :STATus:OPERation:PRESet
- :STATus:QUEStionable:CONDition
- :STATus:QUEStionable:CURRent:ENABle
- :STATus:QUEStionable:CURRent[:EVENt]
- :STATus:QUEStionable:DCONverter:ENABle
- :STATus:QUEStionable:DCONverter[:EVENt]

- :STATus:QUEStionable:ENABle
- :STATus:QUEStionable:EVENt
- :STATus:QUEStionable:LIMit:CONDition
- :STATus:QUEStionable:LIMit:ENABle
- :STATus:QUEStionable:LIMit[:EVENt]
- :STATus:QUEStionable:Limit:AM[1-1]:CONDition
- :STATus:QUEStionable:LIMit:AM[1-1]:ENABle
- :STATus:QUEStionable:LIMit:AM[1-1][:EVENt]
- :STATus:QUEStionable:LIMit:AM[1-1]:NTRansition
- :STATus:QUEStionable:LIMit:AM[1-1]:PTRansition
- :STATus:QUEStionable:Limit:BB[1-1]:CONDition
- :STATus:QUEStionable:LIMit:BB[1-1]:ENABle
- :STATus:QUEStionable:LIMit:BB[1-1][:EVENt]
- :STATus:QUEStionable:LIMit:BB[1-1]:NTRansition
- :STATus:QUEStionable:LIMit:BB[1-1]:PTRansition
- :STATus:QUEStionable:LIMit:FP[1-1]:CONDition
- :STATus:QUEStionable:LIMit:FP[1-1]:ENABle
- :STATus:QUEStionable:LIMit:FP[1-1][:EVENt]
- :STATus:QUEStionable:LIMit:FP[1-1]:NTRansition
- :STATus:QUEStionable:LIMit:FP[1-1]:PTRansition
- :STATus:QUEStionable:LIMit:NTRansition
- :STATus:QUEStionable:LIMit:PN[1-1]:CONDition
- :STATus:QUEStionable:LIMit:PN[1-1]:ENABle
- :STATus:QUEStionable:LIMit:PN[1-1][:EVENt]
- :STATus:QUEStionable:LIMit:PN[1-1]:NTRansition
- :STATus:QUEStionable:LIMit:PN[1-1]:PTRansition

- :STATus:QUEStionable:LIMit:PS[1-1]:CONDition
- :STATus:QUEStionable:LIMit:PS[1-1]:ENABle
- :STATus:QUEStionable:LIMit:PS[1-1][:EVENt]
- :STATus:QUEStionable:LIMit:PS[1-1]:NTRansition
- :STATus:QUEStionable:LIMit:PS[1-1]:PTRansition
- :STATus:QUEStionable:LIMit:PTRansition
- :STATus:QUEStionable:LIMit:SP[1-1]:CONDition
- :STATus:QUEStionable:LIMit:SP[1-1]:ENABle
- :STATus:QUEStionable:LIMit:SP[1-1][:EVENt]
- :STATus:QUEStionable:LIMit:SP[1-1]:NTRansition
- :STATus:QUEStionable:LIMit:SP[1-1]:PTRansition
- :STATus:QUEStionable:LIMit:TR[1-1]:CONDition
- :STATus:QUEStionable:LIMit:TR[1-1]:ENABle
- :STATus:QUEStionable:LIMit:TR[1-1][:EVENt]
- :STATus:QUEStionable:LIMit:TR[1-1]:NTRansition
- :STATus:QUEStionable:LIMit:TR[1-1]:PTRansition
- :STATus:QUEStionable:LIMit:USER[1-1]:CONDition
- :STATus:QUEStionable:LIMit:USER[1-1]:ENABle
- :STATus:QUEStionable:LIMit:USER[1-1][:EVENt]
- :STATus:QUEStionable:LIMit:USER[1-1]:NTRansition
- :STATus:QUEStionable:LIMit:USER[1-1]:PTRansition
- :STATus:QUEStionable:MISC:ENABle
- :STATus:QUEStionable:MISC[:EVENt]
- :STATus:QUEStionable:NTRansition
- :STATus:QUEStionable:PHASe:ENABle
- :STATus:QUEStionable:PHASe[:EVENt]

- :STATus:QUEStionable:POWer:ENABle
- :STATus:QUEStionable:POWer[:EVENt]
- :STATus:QUEStionable:PTRansition
- :STATus:QUEStionable:REFerence:ENABle
- :STATus:QUEStionable:REFerence[:EVENt]

Other topics about Status Reporting System

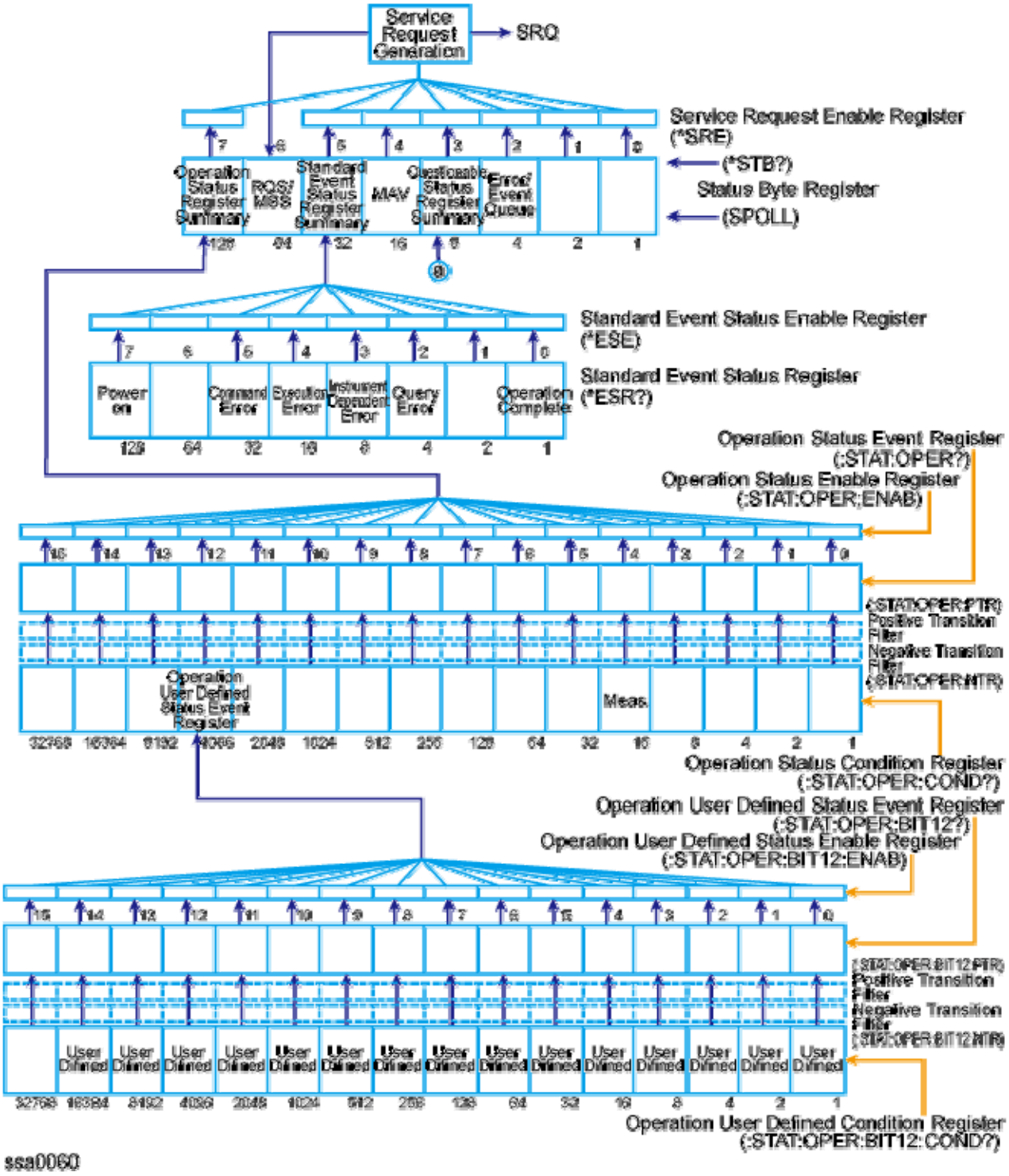
Status Register Structure

Status Register Structure

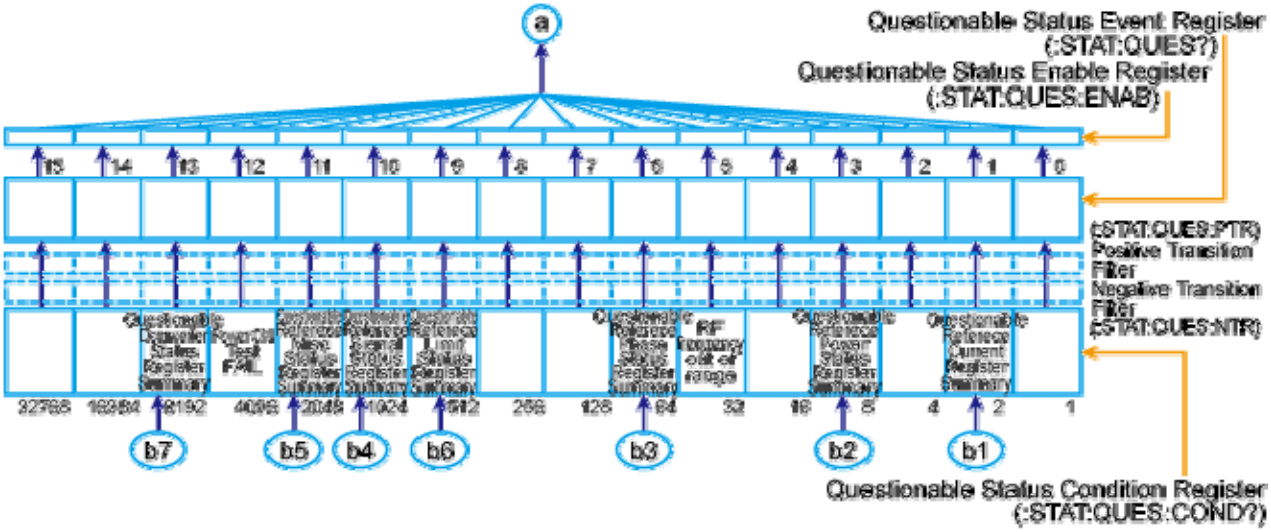
This section describes the E5052B's status registers in each hierarchy.

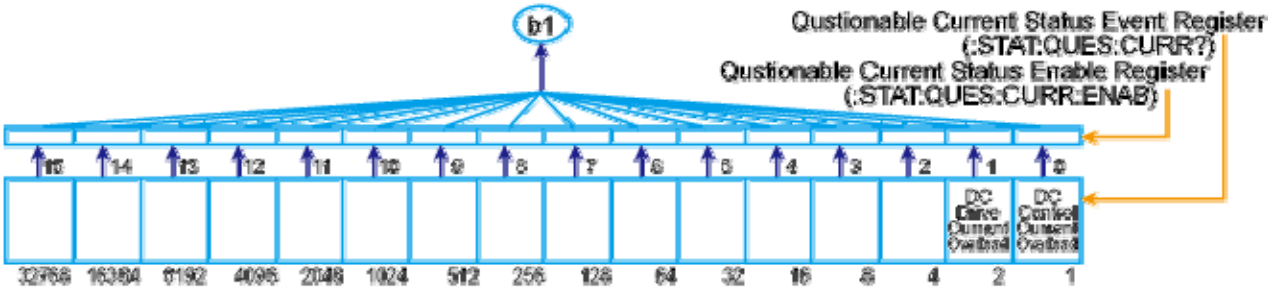
- Status Register Structure 1
- Status Register Structure 2
- Status Register Structure 3
- Status Register Structure 4
- Status Register Structure 5
- Status Register Structure 6
- Status Register Structure 7

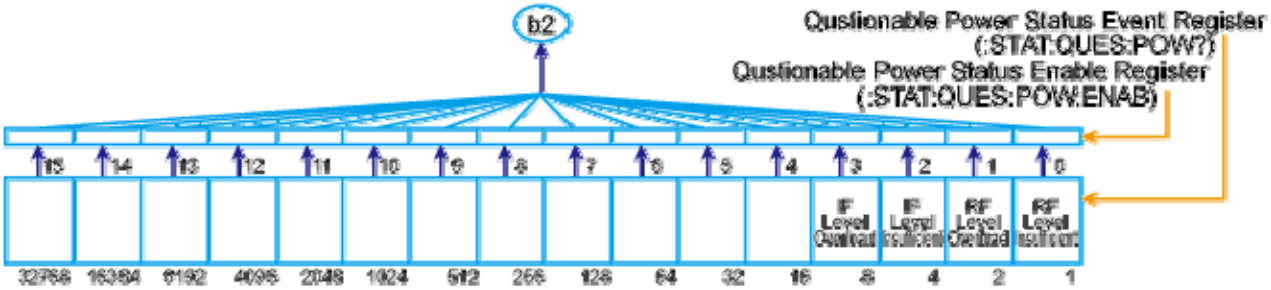
Status Register Structure 1



Status Register Structure 2

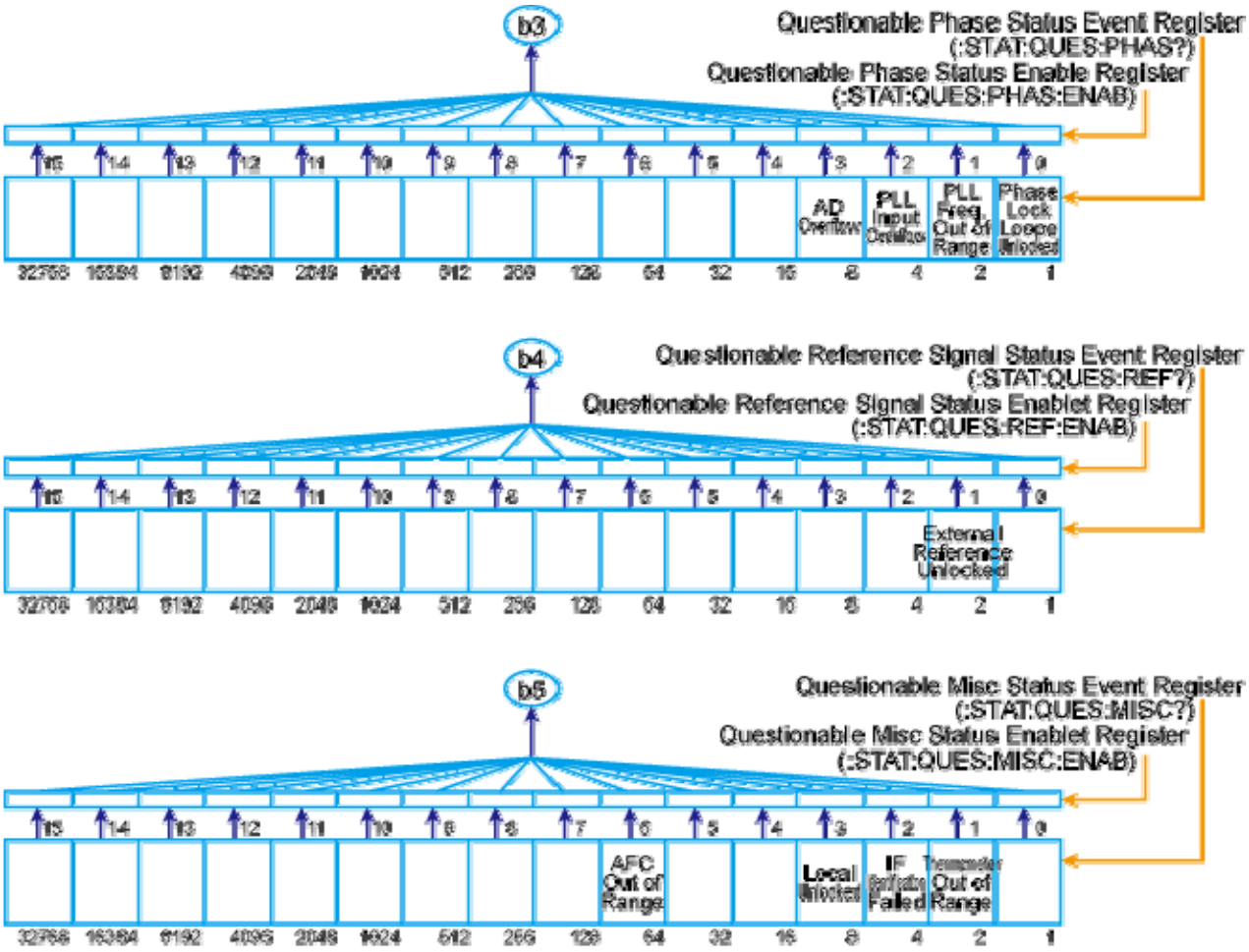






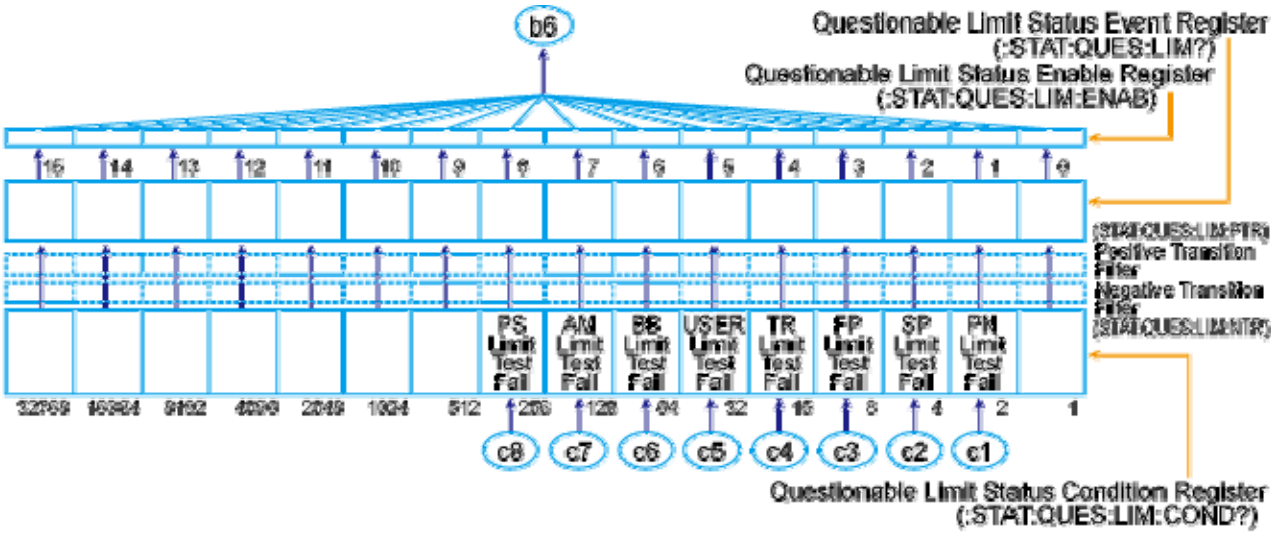
ssa0061

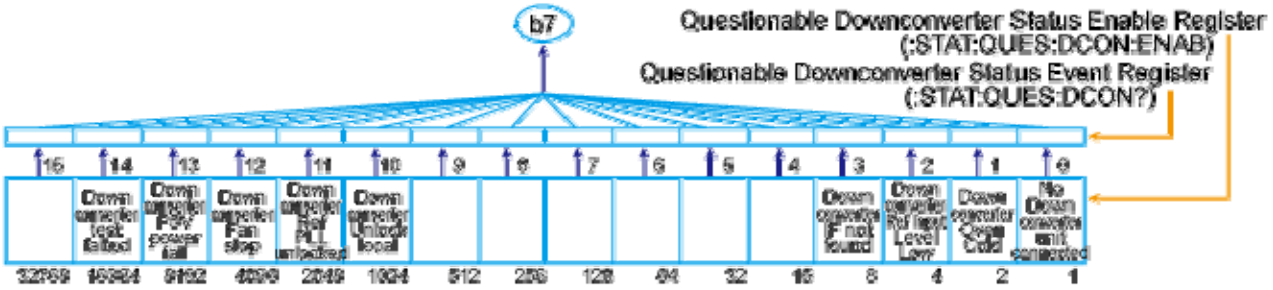
Status Register Structure 3



ssa0062

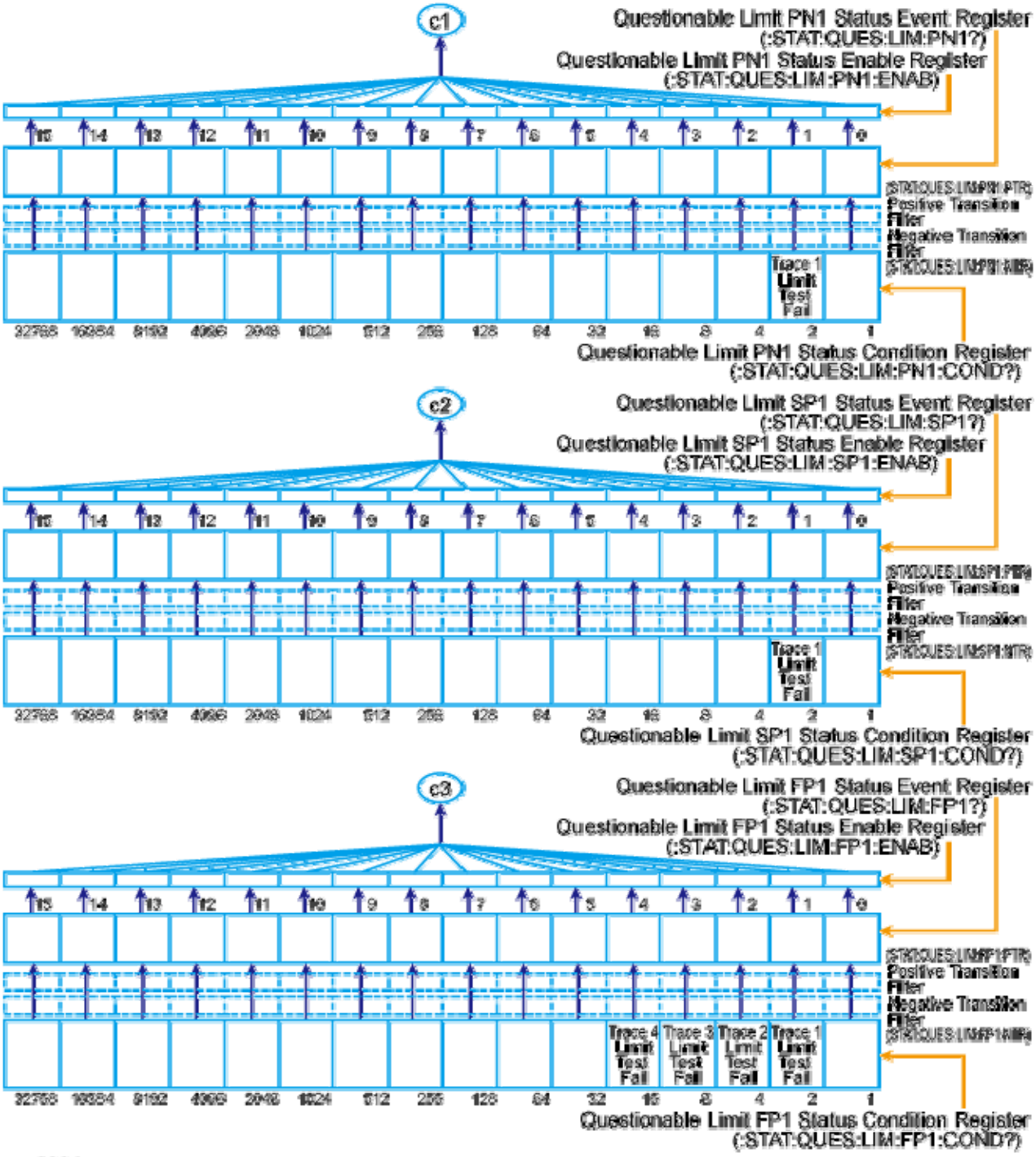
Status Register Structure 4



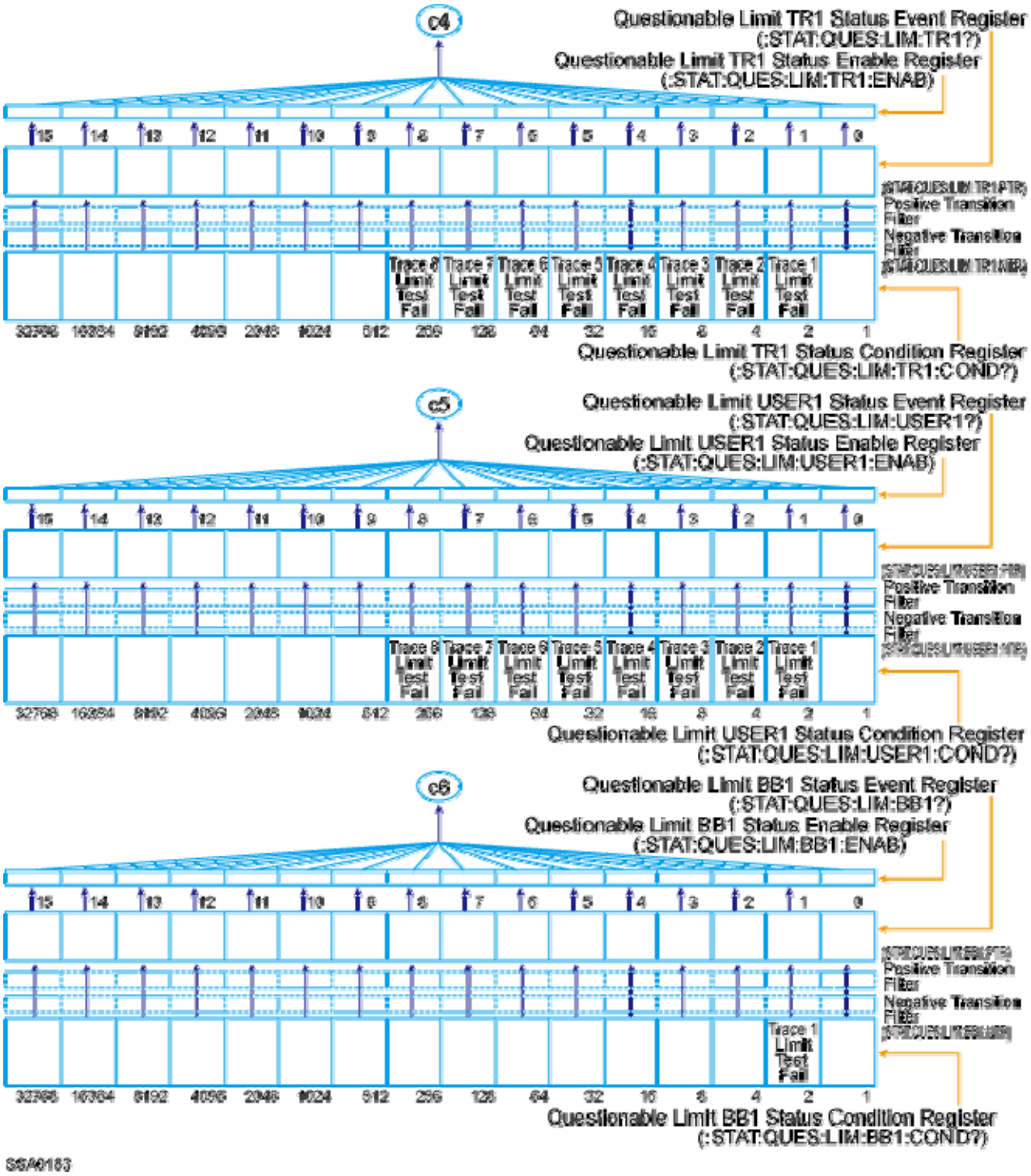


SSA0182

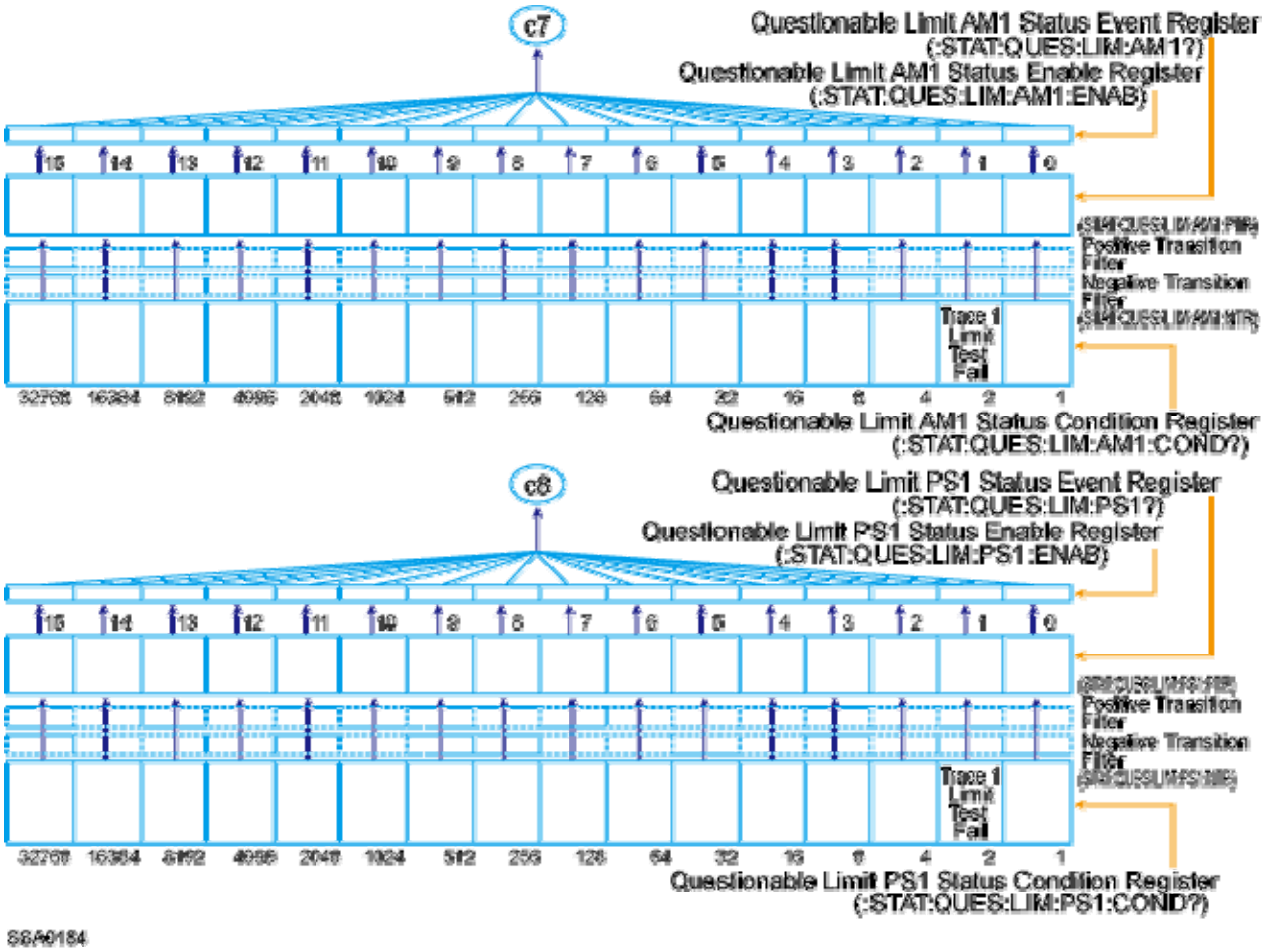
Status Register Structure 5



Status Register Structure 6



Status Register Structure 7



Status Bit Definition

Status Bit Definition

- Status Byte Register
- Standard Event Status Register (ESR)
- Operation Status Condition Register
- Operation User Defined Status Condition Register
- Questionable Status Condition Register
- Questionable Current Status Event Register
- Questionable Power Status Event Register
- Questionable Phase Status Event Register
- Questionable Reference Signal Status Event Register
- Questionable Misc Status Event Register
- Questionable Limit Status Event Register
- Questionable Limit PN1 Status Event Register
- Questionable Limit SP1 Status Event Register
- Questionable Limit FP1 Status Event Register
- Questionable Limit TR1 Status Event Register
- Questionable Limit USER1 Status Event Register
- Questionable Limit BB1 Status Event Register
- Questionable Limit AM1 Status Event Register
- Questionable Limit PS1 Status Event Register
- Questionable Downconverter Status Event Register

Status Byte Register

Bit Position	Name	Description
0-1	Not used	Always 0.
2	Error/Event Queue	Set to “1” if error/event queue contains data; reset to "0” when all data has been retrieved.
3	Questionable Status Register Summary	Set to “1” when one of the enabled bits in status event status register is set to “1.”
4	MAV (Message Available)	Set to “1” when output queue contains data; reset to “0” when all data has been retrieved.
5	Standard Event Status Register Summary	Set to “1” when one of the enabled bits in status event status register is set to “1.”
6	RQS/MSS	Set to “1” when any of the status byte register bits enabled by service request enable register is set to “1”; reset to "0” when all data has been retrieved through serial polling. See IEEE 488.1 and IEEE 488.2 standards for details.
7	Operation Status Register Summary	Set to “1” when one of the enabled bits in operational status register is set to “1.”

NOTE Issuing the *CLS command will clear all of the bits from the status byte register.

Standard Event Status Register (ESR)

Bit Position	Name	Description
0	Operation Complete	Always 1.

1	Not used	Always 0.
2	Query Error	1. Set to “1” when E5052B receives a data output request but there is no data to output. 2. Set to “1” when the data of E5052B's output queue has been cleared for a new message received before completion of data output.
3	Instrument Dependent Error	Set to “1” for an error that is not a command, query, or execution error.
4	Execution Error	1. Set to “1” when any parameter in an SCPI command exceeds its input range or is inconsistent with E5052B's capabilities. 2. Set to “1” when an SCPI command cannot be properly executed due to some condition of E5052B
5	Command Error	1. Set to “1” when an IEEE 488.2 syntax error occurs (a command sent to E5052B does not follow the IEEE 488.2 syntax). Possible violations include command parameters violating E5052B listening formats or other unacceptable conditions. 2. Set to “1” when a semantic error occurs. Possible causes include sending to E5052B a command containing misspellings or an IEEE 488.2 command not supported by E5052B. 3. Set to “1” when GET (Group Execution Trigger) is input while receiving a program message.
6	Not used	Always 0.
7	Power ON	Set to “1” when the E5052B is powered ON, or when the firmware is restarted.

NOTE Issuing the *CLS command will clear all of the bits from the standard event status register

Operation Status Condition Register

Bit Position	Name	Description
0-3	Not used	Always 0.
4	Measurement	Set to “1” during measurement.
5-11	Not used	Always 0.
12	Operation user defined status register summary	Set to “1” when one of the enable bits in operation user defined status register is set to “1.”
13-15	Not used	Always 0.

NOTE Bit4 is set to “1” even in the “Waiting for Trigger” state when the trigger is set to “Ext/Video” in the transient measurement. This is because pre-triggering is performed in “Waiting for Trigger” state in the transient measurement.

NOTE Issuing the *CLS command will clear all bits from the operation status event register

Operation User Defined Status Condition Register

Bit Position	Name	Description
0-14	Defined by the user	Set to “1” as defined by the user.
15	Not used	Always 0.

Questionable Status Condition Register

Bit Position	Name	Description
0	Not used	Always 0.
1	Questionable Current Status Register Summary	Set to “1” when one of the enable bits in questionable current status register is set to “1.”
2	Not used	Always 0.
3	Questionable Power Status Register Summary	Set to “1” when one of the enable bits in questionable power status register is set to “1.”
4	Not used	Always 0.
5	RF frequency out of range	Set to “1” when DUT’s oscillation frequency is out of E5052B’s measurement range. In this case, DUT’s oscillation frequency should be verified.
6	Questionable Phase Status Register Summary	Set to “1” when one of the enable bits in the questionable phase status register is set to “1.”
7-8	Not used	Always 0.
9	Questionable Limit Status Register Summary	Set to “1” when one of the enable bits in the questionable limit status register is set to “1.”
10	Questionable Reference Signal Status	Set to “1” when one of the enable bits in the questionable reference signal status register is set to “1.”

	Register Summary	
11	Questionable Misc Status Register Summary	Set to “1” when one of the enable bits in the questionable misc. status register is set to “1.”
12	Power-on Test FAIL	Set to “1” when the power-on self test results in “FAIL.”
13-15	Not used	Always 0.

NOTE Issuing the *CLS command will clear all of the bits from the questionable status condition register.

Questionable Current Status Event Register

Bit Position	Name	Description
0	DC Control Current Overloaded	Set to “1” when excessive DC control current is loaded.
1	DC Power Current Overloaded	Set to “1” when excessive DC power current is loaded.
2-15	Not used	Always 0.

NOTE Issuing the *CLS command will clear all of the bits from the questionable current status event register.

Questionable Power Status Event Register

Bit Position	Name	Description
0	RF Level Insufficient	Set to “1” when an insufficient level of RF input is applied.
1	RF Level	Set to “1” when an excessive level of RF input

	Overloaded	is applied.
2	Insufficient *IF* Level	Set to “1” when an insufficient level of IF input is applied.
3	IF Level Overloaded	Set to “1” when an excessive level of IF input is applied.
4-15	Not used	Always 0.

NOTE Issuing the *CLS command will clear all of the bits in the questionable power status event register.

Questionable Phase Status Event Register

Bit Position	Name	Description
0	Phase Lock Loop Unlocked	Set to “1” when Phase Lock Loop is not locked.
1	PLL Frequency Out of Range	Set to “1” when PLL frequency is out of E5052B’s measurement range.
2	PLL Input Overflow	Set to “1” when PLL input overflows.
3	A/D Overflow	Set to “1” when signal level saturates at A/D.
4-15	Not used	Always 0.

NOTE Issuing the *CLS command will clear all of the bits in the questionable phase status event register.

Questionable Reference Signal Status Event Register

Bit Position	Name	Description
0	Not used	Always 0.
1	External Reference Unlock	Set to “1” when external reference is not locked.

2-15	Not used	Always 0.

NOTE Issuing the *CLS command will clear all of the bits in the questionable reference signal status event register.

Questionable Misc Status Event Register

Bit Position	Name	Description
0	Not used	Always 0.
1	Thermometer Out of Range	Set to “1” when thermometer is out of range.
2	*IF* not Detected	Set to “1” when IF is not detected.
3	Local Unlock	Set to “1” when local oscillator is unlocked.
4-5	Not used	Always 0.
6	AFC Out of Loop	Set to “1” when an AFC out of loop error is applied.
7-15	Not used	Always 0.

NOTE Issuing the *CLS command will clear all of the bits in the questionable misc status event register.

Questionable Limit Status Event Register

Bit Position	Name	Description
0	Not used	Always 0.
1	Questionable Limit PN1 Status Register Summary	Set to “1” when one of the enable bits in questionable limit PN1 status register is set to “1.”
2	Questionable Limit SP1	Set to “1” when one of the enable bits in questionable limit SP1 status register is set to

	Status Register Summary	“1.”
3	Questionable Limit FP1 Status Register Summary	Set to “1” when one of the enable bits in questionable limit FP1 status register is set to “1.”
4	Questionable Limit TR1 Status Register Summary	Set to “1” when one of the enable bits in questionable limit TR1 status register is set to “1.”
5	Questionable Limit USER1 Status Register Summary	Set to “1” when one of the enable bits in questionable limit USER1 status register is set to “1.”
6	Questionable Limit BB1 Status Register Summary	Set to “1” when one of the enable bits in questionable limit BB1 status register is set to “1.”
7	Questionable Limit AM1 Status Register Summary	Set to “1” when one of the enable bits in questionable limit AM1 status register is set to “1.”
8	Questionable Limit PS1 Status Register Summary	Set to “1” when one of the enable bits in questionable limit PS1 status register is set to “1.”
6-15	Not used	Always 0.

NOTE Issuing the *CLS command will clear all of the bits in the questionable limit status event

Questionable Limit AM1 Status Event Register

Bit Position	Name	Description
0	Not used	Always 0.
1	Trace 1 Limit Test Fail	Set to “0” when a measurement cycle begins; set to “1” when the measurement cycle finishes and returns “fail” as the limit test result for trace 1.
2-15	Not used	Always 0.

NOTE Issuing the *CLS command will clear all of the bits in the questionable limit AM1 status event register.

Questionable Limit BB1 Status Event Register

Bit Position	Name	Description
0	Not used	Always 0.
1	Trace 1 Limit Test Fail	Set to “0” when a measurement cycle begins; set to “1” when the measurement cycle finishes and returns “fail” as the limit test result for trace 1.
2-15	Not used	Always 0.

NOTE Issuing the *CLS command will clear all of the bits in the questionable limit BB1 status event register.

Questionable Limit PN1 Status Event Register

Bit Position	Name	Description
0	Not used	Always 0.
1	Trace 1 Limit Test Fail	Set to “0” when a measurement cycle begins; set to “1” when the measurement cycle finishes and returns “fail” as the limit test result for trace 1.
2-15	Not used	Always 0.

NOTE Issuing the *CLS command will clear all of the bits in the questionable limit PN1 status event register.

Questionable Limit SP1 Status Event Register

Bit Position	Name	Description
0	Not used	Always 0.
1	Trace 1 Limit Test Fail	Set to “0” when a measurement cycle begins; set to “1” when the measurement cycle finishes and returns “fail” as the limit test result for trace 1.
2-15	Not used	Always 0.

NOTE Issuing the *CLS command will clear all of the bits in the questionable limit SP1 status event register.

Questionable Limit PS1 Status Event Register

Bit Position	Name	Description
0	Not used	Always 0.
1	Trace 1 Limit Test Fail	Set to “0” when a measurement cycle begins; set to “1” when the measurement cycle finishes and returns “fail” as the limit test result for trace 1.
2-15	Not used	Always 0.

NOTE Issuing the *CLS command will clear all of the bits in the questionable limit PS1 status event register.

Questionable Limit FP1 Status Event Register

Bit Position	Name	Description
0	Not used	Always 0.
1	Trace 1 Limit Test Fail	Set to “0” when a measurement cycle begins; set to “1” when the measurement cycle finishes and returns “fail” as the limit test result for trace 1.
2	Trace 2 Limit Test Fail	Set to “0” when a measurement cycle begins; set to “1” when the measurement cycle finishes and returns “fail” as the limit test result for trace 2.
3	Trace 3 Limit Test Fail	Set to “0” when a measurement cycle begins;set to “1” when the measurement cycle finishes and returns “fail” as the limit test result for trace 3.
4	Trace	Set to “0” when a measurement cycle begins; set to “1”

	4 Limit Test Fail	when the measurement cycle finishes and returns “fail” as the limit test result for trace 4.
5-15	Not used	Always 0.

NOTE Issuing the *CLS command will clear all of the bits in the questionable limit FP1 status event register

Questionable Limit TR1 Status Event Register

Bit Position	Name	Description
0	Not used	Always 0.
1	Trace 1 Limit Test Fail	Set to “0” when a measurement cycle begins; set to “1” when the measurement cycle finishes and returns “fail” as the limit test result for trace 1.
2	Trace 2 Limit Test Fail	Set to “0” when a measurement cycle begins; set to “1” when the measurement cycle finishes and returns “fail” as the limit test result for trace 2.
3	Trace 3 Limit Test Fail	Set to “0” when a measurement cycle begins;set to “1” when the measurement cycle finishes and returns “fail” as the limit test result for trace 3.
4	Trace 4 Limit Test Fail	Set to “0” when a measurement cycle begins; set to “1” when the measurement cycle finishes and returns “fail” as the limit test result for trace 4.
5	Not used	Always 0.
6	Trace 6 Limit	Set to “0” when a measurement cycle begins; set to “1” when the measurement cycle finishes and returns “fail”

	Test Fail	as the limit test result for trace 6.
7	Trace 7 Limit Test Fail	Set to “0” when a measurement cycle begins; set to “1” when the measurement cycle finishes and returns “fail” as the limit test result for trace 7.
8	Trace 8 Limit Test Fail	Set to “0” when a measurement cycle begins; set to “1” when the measurement cycle finishes and returns “fail” as the limit test result for trace 8.
9-15	Not used	Always 0.

NOTE Issuing the *CLS command will clear all of the bits in the questionable limit TR1 status event register.

Questionable Limit USER1 Status Event Register

Bit Position	Name	Description
0	Not used	Always 0.
1	Trace 1 Limit Test Fail	Set to “0” when a measurement cycle begins; set to “1” when the measurement cycle finishes and returns “fail” as the limit test result for trace 1.
2	Trace 2 Limit Test Fail	Set to “0” when a measurement cycle begins; set to “1” when the measurement cycle finishes and returns “fail” as the limit test result for trace 2.
3	Trace 3 Limit Test Fail	Set to “0” when a measurement cycle begins;set to “1” when the measurement cycle finishes and returns “fail” as the limit test result for trace 3.
4	Trace	Set to “0” when a measurement cycle begins; set to “1”

	4 Limit Test Fail	when the measurement cycle finishes and returns “fail” as the limit test result for trace 4.
5	Trace 5 Limit Test Fail	Set to “0” when a measurement cycle begins; set to “1” when the measurement cycle finishes and returns “fail” as the limit test result for trace 5.
6	Trace 6 Limit Test Fail	Set to “0” when a measurement cycle begins;set to “1” when the measurement cycle finishes and returns “fail” as the limit test result for trace 6.
7	Trace 7 Limit Test Fail	Set to “0” when a measurement cycle begins;set to “1” when the measurement cycle finishes and returns “fail” as the limit test result for trace 7.
8	Trace 8 Limit Test Fail	Set to “0” when a measurement cycle begins;set to “1” when the measurement cycle finishes and returns “fail” as the limit test result for trace 8.
9 - 15	Not used	Always 0.

NOTE Issuing the *CLS command will clear all of the bits in the questionable limit USER1 status event register.

Questionable Downconverter Status Event Register

Bit Position	Name	Description
0	Downconverter not connected	Set to 1 when connection to the downconverter is not detected.
1	Downconverter Oven not ready	Set to 1 when the oven is not ready.
2	Downconverter Ref Input Level Low	Set to 1 when the 10-MHz reference signal level is low.

3	Downconverter *IF* not found	Set to 1 when the downconverter IF is not found.
4-9	Not used	Always 0.
10	Downconverter unlocked	Set to 1 when the downconverter cannot be locked.
11	Downconverter Ref PLL unlocked	Set to 1 when the PLL circuit of the downconverter is unlocked during measurement.
12	Downconverter Fan Stop	Set to 1 when the fan in the downconverter stops.
13	Downconverter P5V power failure	Set to 1 when the 5-V power to the downconverter fails.
14	Downconverter test failed	Set to 1 when the self-test of the downconverter fails.
15	Not used	Always 0.

NOTE Issuing the *CLS command will clear all of the bits in the Questionable Limit USER1 Status Event Register.

Working with Automatic Test Systems

- Detecting Occurrence of an Error
- Improving Command Processing Speed
- Limit Test
- Preventing Erroneous Key Operation on the Front Panel (Key Lock feature)

Detecting Occurrence of an Error

- Using the Status Reporting System
- Using the Error Queue
- Example of Error_Detection using an SRQ

Other topics about Working with Automatic Test Systems

Using the Status Reporting System

The status of the E5052B can be monitored through the status registers. This section describes how to detect an error using the status registers. For a complete description of the status report mechanism, including the specifications of each bit, see Status Reporting System.

The occurrence of an error will be shown in the standard event status register. An SRQ (service request) is useful when you create a program that uses the information reported by this register to detect the occurrence of errors.

To detect the an error via an SRQ, use one of the following commands:

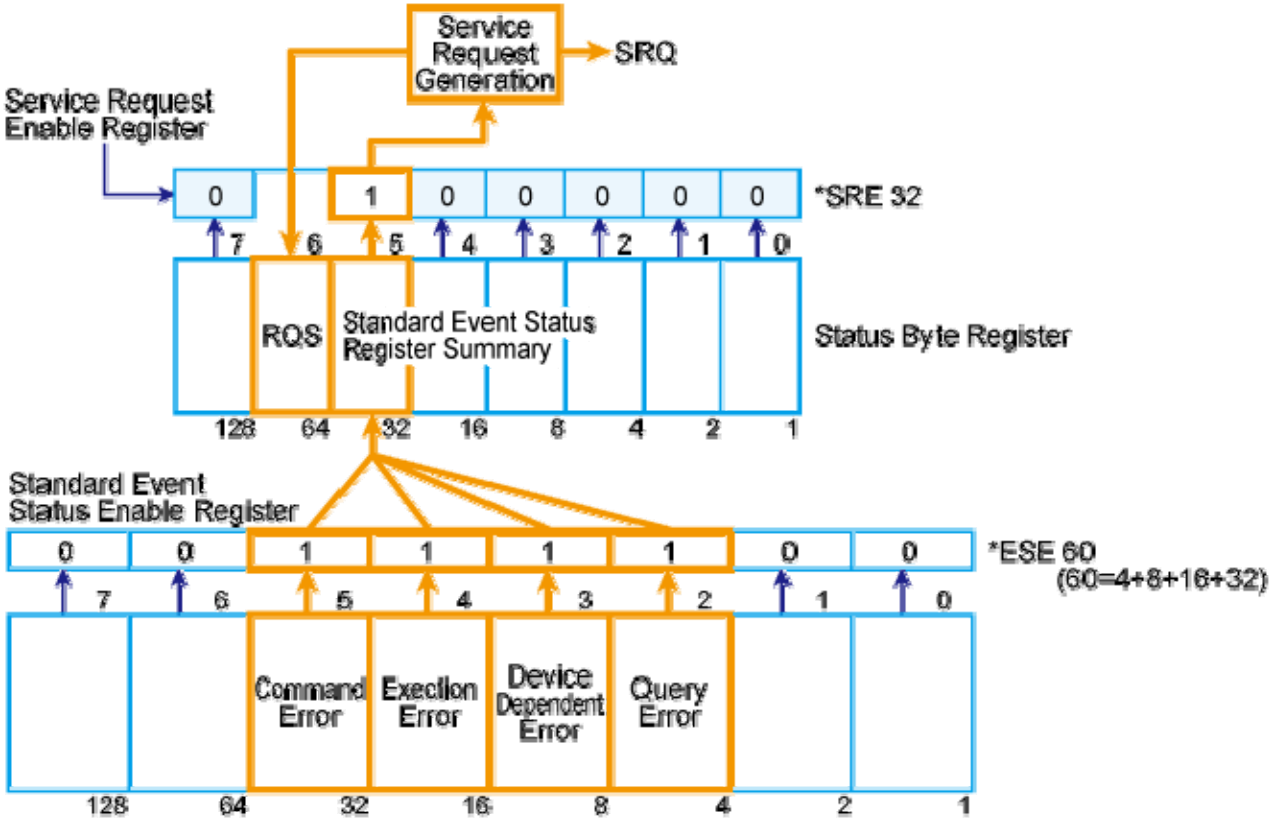
- *SRE
- *ESE

Follow these steps:

Set the E5052B so that it generates an SRQ when any of the error occurrence bits is set to 1 in the standard event status register.

When an SRQ is generated, the program interrupts the measurement cycle.

SRQ Generation Sequence (when an error occurs)



ssa0057

Using the Error Queue

An error queue holds the number for the error and the error message. Reading the error queue allows the user to verify the error that has occurred. To retrieve the content of the error queue, use :SYSTem:ERRor[:NEXT]

The error queue can be used in the following ways:

1. It is used as a branch for error handling. When an error queue is retrieved, it returns 0 as the error number and &dlq;No error&drq; as the error message if no error is detected. This can be used for detecting an error and for branching the flow of a program. This is also useful for handling a specific error(s). Note that using this method prevents the user from performing any processing during the occurrence of an error.
2. When an error is detected using SRQ, the error queue is used to examine the error.

Example of error detection using an SRQ is a sample program that demonstrates how to use an SRQ to detect the occurrence of an error.

This program sets the SRQs and then intentionally sets a wrong trigger to generate an error, which is handled by the program. In the error handling part, this program examines the error and then displays the error number and error message.

Example of Error Detection using an SRQ

```
110 DIM Buff$[9],Err_msg$[100]
120 INTEGER Err_no
130 !
140 ASSIGN @Agte5052 TO 717
150 !
160 OUTPUT @Agte5052;"*ESE 60"
170 OUTPUT @Agte5052;"*SRE 32"
180 OUTPUT @Agte5052;"*CLS"
190 OUTPUT @Agte5052;"*OPC?"
200 ENTER @Agte5052;Buff$
210 !
220 ON INTR 7 GOTO Err_proc
230 ENABLE INTR 7;2
240 OUTPUT @Agte5052;":TRIG:MODE PN1"
250 OUTPUT @Agte5052;":TRIG:SP:SOUR BUS"
260 OUTPUT @Agte5052;":INIT:SP:CONT ON"
280 OUTPUT @Agte5052;"*TRG"
290 PRINT "Waiting..."
300 GOTO Skip_err
310 Err_proc: OFF INTR 7
320 OUTPUT @Agte5052;":SYST:ERR?"
330 ENTER @Agte5052;Err_no,Err_msg$
340 PRINT "Error occured."
350 PRINT "No:";Err_no,"Description: "&Err_msg$
360 PRINT "Program Interrupt."
370 GOTO Prog_end
```

```
380 Skip_err: PRINT "Program Done"
390 Prog_end: END
```

Improving Command Processing Speed

- Overview
- When Measurement Results (trace) do not needs Updation
- When Measurement Results (trace) need Updation

Other topics about Working with Automatic Test Systems

Overview

SCPI commands should be processed quickly to improve throughput when the commands are frequently executed (for example, reading out the trace for each measurement).

With the E5052B, the processing time for SCPI commands can be improved by decreasing the refresh rate of the LCD display.

When Measurement Results (trace) do not need Updation

When the measurement trace does not need to be updated, turn off the update function of the LCD display. This improves the processing speed of SCPI commands and eliminates the updating time needed by the screen.

To turn off the update function of the LCD display, use the following command:

- :DISPlay:ENABle

When Measurement Results (trace) need Updation

The measurement trace can be updated by using following command with the update of the LCD display turned off.

- :DISPlay:UPDate:IMMediate

The update of the LCD display is off even if you execute this command.

Turn Off the update of the LCD display.

- :DISPlay:ENABle

Execute all SCPI commands required before measurement, including settings of measurement conditions.

Perform the measurement.

Execute the commands for reading out or analyzing the measurement result. Note that reading out the result in binary format accelerates data transfer.

Execute the following command to update the LCD display once.

- :DISPlay:UPDate:IMMediate

NOTE When the update of the LCD display is off, “Update Off” is displayed in the instrument status bar.

Limit Test

- Overview
- Reading Limit Lines from Files
- Using Commands to define Limit Lines

Other topics about Working with Automatic Test Systems

Overview

This section describes how to define the limit lines and determine pass or fail with regard to the limit test function. For more information on the concept of the limit test, refer to Data Analysis and Result Output.

Using Commands to define Limit Lines

You can define the limit lines by specifying any limit value to the parameter of SCPI commands.

The program is described in detail below:

Line 120

Define the data array (variable) for the limit lines.

Line 170 to 240

When bit 4 of the operation status condition register changes from “1” to “0” (negative transition), bit 4 of the operation status event register is set to “1”.

Line 260 to 280

Set the upper and lower limit values for the limit lines in the array.

Line 310 to 320

Specify the segment number for the upper and lower limit values.

Line 330 to 340

Specify the upper and lower limit values on the trace.

Line 360 to 400

Trigger the instrument.

Line 420 to 450

It repeats to the measurement end.

Line 460

Display the limit lines.

Line 470

Display the determination result. (For fail only.)

Line 480

Activate the limit test function.

Using Commands to define Limit Lines

```
110 DIM Buff$[9]
120 REAL Udata(1:8),Ldata(1:4)
130 INTEGER St
140 !
150 ASSIGN @Agte5052 TO 717
160 !
170 OUTPUT @Agte5052;"*ESE 60"
180 OUTPUT @Agte5052;"*SRE 32"
190 OUTPUT @Agte5052;":STAT:OPER:PRT 0"
200 OUTPUT @Agte5052;":STAT:OPER:NTR 16"
210 OUTPUT @Agte5052;":STAT:OPER:ENAB 16"
220 OUTPUT @Agte5052;"*CLS"
230 OUTPUT @Agte5052;"*OPC?"
240 ENTER @Agte5052;Buff$
250 !
260 READ Udata(*), Ldata(*)
270 DATA 0,1.4E+9,2.E-5,1.4E+9,2.E-5,1.6E+9,1.E-4,1.6E+9
280 DATA 0,1.2E+9,1.E-4,1.2E+9
290 !
300 OUTPUT @Agte5052;":DISP:WIND:ACT FP1"
```

```
310 OUTPUT @Agte5052;":CALC:FP:TRAC:LIM:UPP:SEGM:COUN 2"
320 OUTPUT @Agte5052;":CALC:FP:TRAC:LIM:LOW:SEGM:COUN 1"
330 OUTPUT @Agte5052;":CALC:FP:TRAC:LIM:UPP:SEGM:DATA ";Udata(*)
340 OUTPUT @Agte5052;":CALC:FP:TRAC:LIM:LOW:SEGM:DATA ";Ldata(*)
350 !
360 OUTPUT @Agte5052;":TRIG:MODE FP1"
370 OUTPUT @Agte5052;":TRIG:FP:SOUR BUS"
380 OUTPUT @Agte5052;":INIT:FP:CONT OFF"
390 OUTPUT @Agte5052;":INIT:FP:IMM"
400 OUTPUT @Agte5052;"*TRG"
410 !
420 REPEAT
430 OUTPUT @Agte5052;":STAT:OPER:COND?"
440 ENTER @Agte5052;St
450 UNTIL BIT(St,4)=0
460 OUTPUT @Agte5052;"DISP:FP:TRAC:LIM:LINE ON"
470 OUTPUT @Agte5052;"DISP:FP:LIM:FSIG ON"
480 OUTPUT @Agte5052;"CALC:FP:TRAC:LIM:STAT ON"
490 PRINT "End"
500 ASSIGN @Agte5052 TO *
510 Prog_end: END
```

Reading Limit Lines from Files

You can create any upper and lower value for limit lines in the specified format in advance, and read the file later to specify the limit lines.

For information about creating upper and lower values for limit lines, refer to the chapter describing “Data analysis and Result Output” in the User’s Guide.

The file must be saved in the CSV format (with the extension *.csv).

The program is described in detail below:

Save the upper limit values together into one file, and the lower limit values into another.

Line 110

Define the data array (variable) for the limit lines.

Line 190 to 260

When bit 4 of the operation status condition register changes from “1” to “0” (negative transition), bit 4 of the operation status event register is set to “1”.

Line 290

Read the upper limit value for the limit lines from file.

Line 300

Read the lower limit value for the limit lines from file.

Line 320 to 360

Trigger the instrument.

Line 380 to 410

It repeats to the measurement end.

Line 430

Display the limit lines.

Line 440

Display the determination result. (For fail only.)

Line 450

Activate the limit test function.

Reading Limit Lines from File

```
110 DIM Ufile$[20],Lfile$[20],Buff$[9]
120 INTEGER St
130 !
140 Ufile$="f:\temP\upper.csv"
150 Lfile$="f:\temP\lower.csv"
160 !
170 ASSIGN @Agte5052 TO 717
180 !
190 OUTPUT @Agte5052;"*ESE 60"
200 OUTPUT @Agte5052;"*SRE 32"
210 OUTPUT @Agte5052;":STAT:OPER:PRT 0"
220 OUTPUT @Agte5052;":STAT:OPER:NTR 16"
230 OUTPUT @Agte5052;":STAT:OPER:ENAB 16"
240 OUTPUT @Agte5052;"*CLS"
```

```
250 OUTPUT @Agte5052;"*OPC?"
260 ENTER @Agte5052;Buff$
270 !
280 OUTPUT @Agte5052;":DISP:WIND:ACT FP1"
290 OUTPUT @Agte5052;":MMEM:FP:TRAC:LOAD:LIM:UPP """&Ufile$&""""
300 OUTPUT @Agte5052;":MMEM:FP:TRAC:LOAD:LIM:LOW """&Lfile$&""""
310 !
320 OUTPUT @Agte5052;":TRIG:MODE FP1"
330 OUTPUT @Agte5052;":TRIG:FP:SOUR BUS"
340 OUTPUT @Agte5052;":INIT:FP:CONT OFF"
350 OUTPUT @Agte5052;":INIT:FP:IMM"
360 OUTPUT @Agte5052;"*TRG"
370 !
380 REPEAT
390 OUTPUT @Agte5052;":STAT:OPER:COND?"
400 ENTER @Agte5052;St
410 UNTIL BIT(St,4)=0
420 !
430 OUTPUT @Agte5052;"DISP:FP:TRAC:LIM:LINE ON"
440 OUTPUT @Agte5052;"DISP:FP:LIM:FSIG ON"
450 OUTPUT @Agte5052;"CALC:FP:TRAC:LIM:STAT ON"
460 PRINT "End"
470 ASSIGN @Agte5052 TO *
480 Prog_end: END
```

Preventing Erroneous Key Operation on the Front Panel (Key Lock feature)

When no operation is required from the front panel controls, the mouse, or the keyboard, disabling these input devices can prevent erroneous operation that might occur due to accidental touching.

To turn on and off Key Locking, use the following commands:

Operation	***Command***
Locking the front panel controls and the keyboard.	:SYSTem:KLOCk:KBD
Locking the mouse and the touch screen.	:SYSTem:KLOCk:MOUSe

Other topics about Working with Automatic Test Systems

VBA Programming

- Introduction to VBA Programming
- Operation Basics of the E5052B's VBA
- Controlling the E5052B
- Controlling Peripherals
- User Defined Window

Introduction to VBA Programming

- Introduction of the E5052B Macro Function
- An Overview of a Control System Based on the Macro Function
- Overview of E5052B COM Object

Introduction of the E5052B Macro Function

- Overview
- Macro Function

Other topics about Introduction to VBA Programming

Overview

The E5052B has a built-in macro function that allows a single instruction to substitute for multiple instructions. You can have the E5052B automatically execute your own macro program that contains a series of VBA (Visual Basic for Application) statements. The macro function allows you to run a variety of applications; you can control not only the E5052B but also various peripherals from your own macro code.

The VBA is based on the VB (Visual Basic) programming language. Although the VBA is similar to the VB, they are not the same. The VBA is decreased some of the VB's features and added characteristic features for each application. The E5052B VBA is added features for controlling the E5052B. For details of difference between the VBA and the VB, refer to Microsoft official guides, and various books on VBA.

For information on the basic operating procedures for the E5052B's VBA, see Operation Basics of the E5052B's VBA. This manual is not meant to be an in-

depth guide to VBA programming basics and the syntax of VBA functions and commands. Such in-depth information is covered in VBA Help, Microsoft official guides, and various books on VBA.

Macro Function

The macro function allows you to control the E5052B itself as well as various peripherals. You can do the following:

1. Automate repetitive tasks
   You can use the E5052B's macro function to combine several processes into one. Automating repetitive tasks provides higher efficiency and eliminates human error. Once you have contained repetitive tasks in Sub procedures, you can later call the procedures from other programs, thus allowing effective reuse of programming assets.

2. Implement a user interface
   The E5052B VBA supports user forms that simplify creating a visual user interface. User forms guide users through common tasks such as performing measurement and entering data, without requiring familiarity with the E5052B, thus minimizing the possibility of human error.

An Overview of a Control System Based on the Macro Function

- Overview
- Implementing a Control System
- Required Equipment
- Control Methods

Other topics about Introduction to VBA Programming

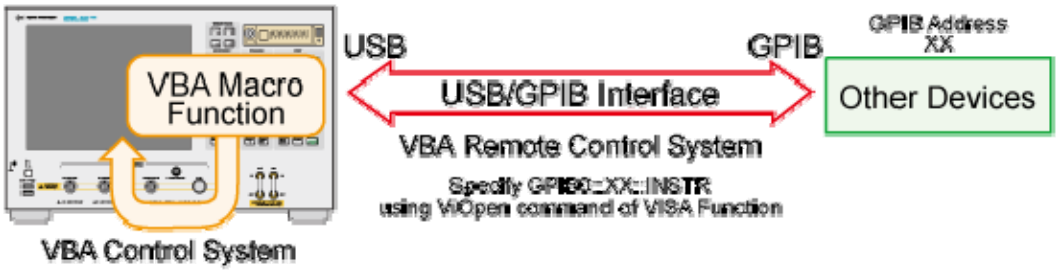
Overview

This section describes how you can use the E5052B's built-in VBA macro function to implement a system that controls the E5052B and peripherals, and what command sets are available for such purposes.

Implementing a Control System

Macro-based control systems are classified into two types: As shown in the following figure, a VBA control system controls the E5052B itself while a VBA remote control system controls peripherals. When you use the macro function to control peripherals, you must connect the E5052B with the peripherals through USB/GPIB interface, USB or LAN, and configure them to communicate over VISA (Virtual Instrument Software Architecture). For information on programming using the VISA library, refer to Programming with VISA.

Configuration example of control system using macro environment



ssa0074

Required Equipment

- E5052B
- Peripherals and/or other instruments that serve your purpose
- USB/GPIB interface, USB Cable, or LAN

Control Methods

The command set you can use differs depending on whether you use the macro function to control the E5052B or a peripheral.

Controlling the E5052B

When you want to control the E5052B itself, you can create a program using COM objects within the E5052B VBA environment. COM objects that come with the E5052B include seven objects specific to the COM interface and COM objects that correspond to SCPI commands.

Controlling a Peripheral

When you want to control a peripheral, you can create a program using VISA library functions within the E5052B VBA environment.

For information on using the VISA library, see Controlling Peripherals. For a complete description of VISA functions, refer to the VISA library's online help.

For information on the GPIB commands available with a particular peripheral, refer to the documentation that comes with the peripheral.

Overview of E5052B COM Object

- Overview
- About COM Object
- Property
- Method
- Event
- Using COM Object to Control E5052B
- Major Control Difference between COM Object and SCPI Command

Other topics about Introduction to VBA Programming

Overview

The E5052B VBA environment provides COM objects that support controlling the E5052B. This section provides an overview of COM objects as well as considerations for using the E5052B's COM objects.

The definitions and specifications of COM are beyond the scope of this guide. Such in-depth information is covered in a variety of books on COM.

About COM Object

When you control the E5052B through the macro function, you can use COM objects as components of your application. The functionality of the E5052B's COM objects is exposed through properties and methods.

Property

A property allows you to read or write a setting or attribute of an object. With the E5052B, you can use properties to set or read the settings of the E5052B.

You can find properties in the list of object types in COM Object Reference.

Method

A method allows you to manipulate an object in a particular way. With the E5052B, you can use methods to perform specific tasks.

You can find methods in the list of object types in COM Object Reference.

Event

An event means an operation from outside that the program can recognize such as clicking a mouse. The E5052B detects events that a specific softkey is pressed using the UserMenu_OnPress(ByVal Key_id As Long) procedure to execute the assigned procedure.

You can find events in the list of object types in COM Object Reference.

Using COM Object to Control E5052B

When you want to control the E5052B, you can use COM objects alone or in conjunction with SCPI commands and the Parse object. The latter method is a little slower than the former method because the Parse object is used to parse the messages of SCPI commands. For instructions on using the E5052B's VBA Editor to create a program that uses COM objects, refer to Operation Basics of the E5052B's VBA.

Major Control Difference between COM Object and SCPI Command

While the control using SCPI commands allows SRQ (Service Request) interrupts through the status reporting mechanism, the control using COM objects does not support SRQ interrupts. Instead of SRQ interrupts, you can use the WaitOnSRQ object to suspend the program until the E5052B is put into the desired state.

Operation Basics of the E5052B's VBA

- Displaying Visual Basic Editor
- Closing Visual Basic Editor
- Switching to the E5052B Measurement Screen
- Making a Preparation Before Coding
- Coding a VBA Program
- Saving a VBA program
- Loading a VBA Program
- Running a VBA Program
- Stopping a VBA Program
- Errors and Debugging
- Printing Output Values in the Echo Window
- Uses Advanced Techniques
- Using VBA Online Help

Displaying Visual Basic Editor

- Overview
- Initial Screen of Visual Basic Editor

Other topics about Operation Basics

Overview

This section describes how to launch Visual Basic Editor.

From the E5052B measurement screen, launch Visual Basic Editor using one of the following methods:

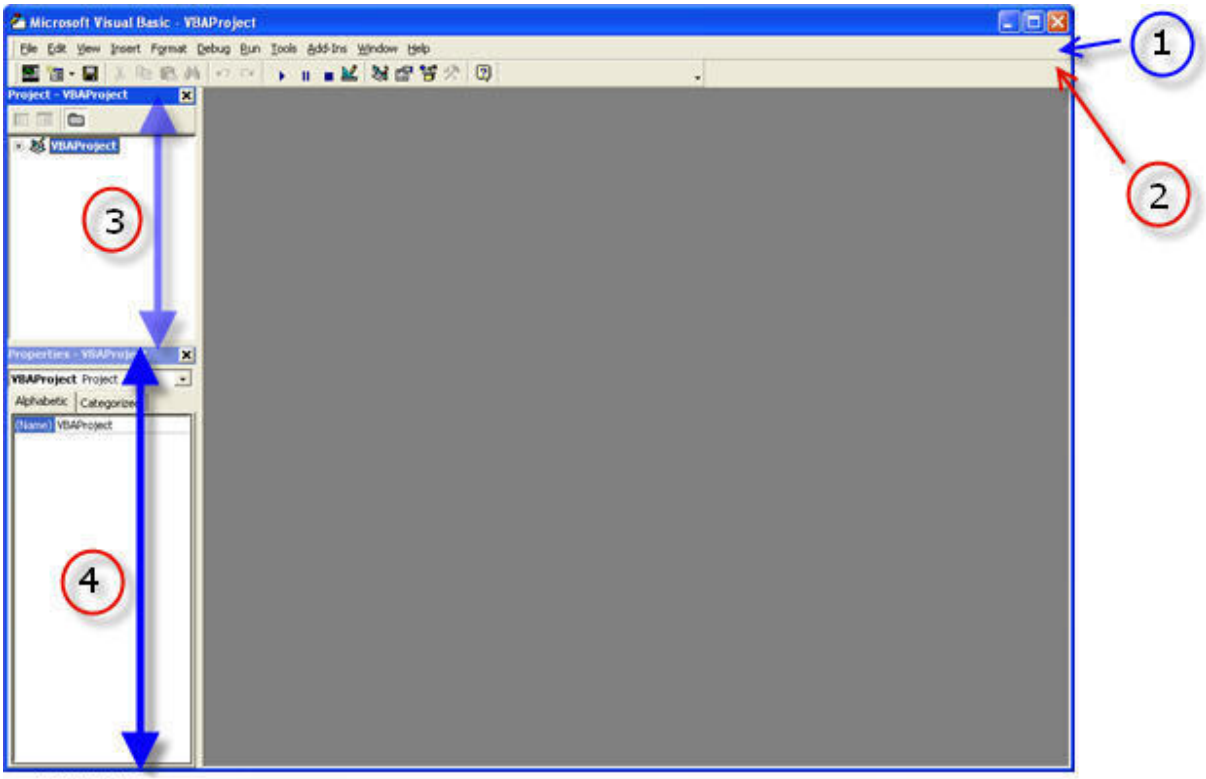
1. **Macro Setup** > **VBA Editor Menu** > **Open Editor**

2. Press **Alt** + **F11** keys on the keyboard.

Initial Screen of Visual Basic Editor

When you launch Visual Basic Editor, it displays the initial screen, which contains a number of windows as shown in the following figure. The initial screen provides the following GUI elements:

Example of Visual Basic Editor initial screen



e5071c119

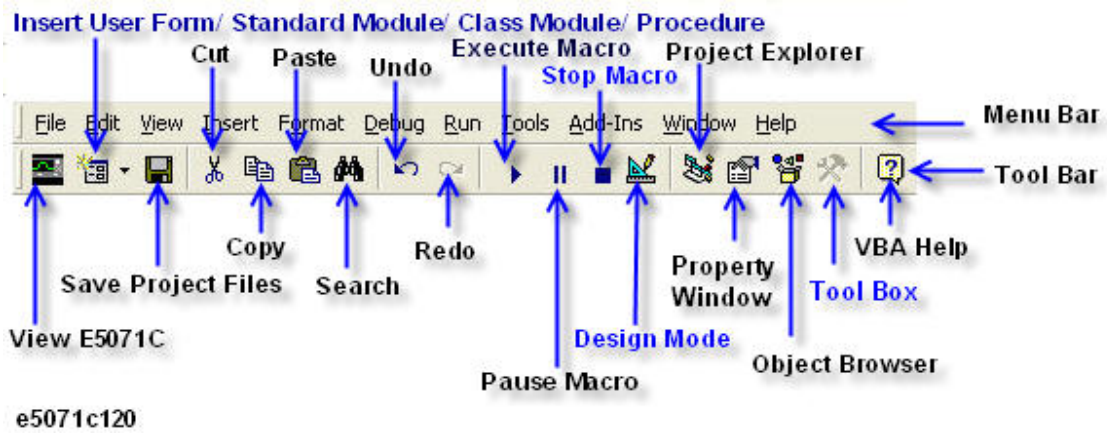
1. Menu bar
2. Toolbar
3. Project Explorer
4. Property Window

Menu Bar

Clicking one of the menu labels brings up the corresponding menu. The menu bar can be used as the primary method to navigate through E5052B's VBA environment.

Toolbar

The toolbar provides access to commonly used commands via icon buttons; these commands are a subset of the commands accessible from the menu bar.



Project Explorer

Within the E5052B's VBA environment, you can develop your application as a project that consists of a number of files (modules). Project Explorer shows a list of all files (modules) that make up a project. The list also includes files (modules) created or loaded in Visual Basic Editor. For information on modules, refer to A Project and Three Types of Module.

To display the project explorer, do one of the following:

1. On the **View** menu, click **Project Explorer**.
2. Press **Ctrl + R** keys on the keyboard.
3. On the toolbar, click .

Property Window

A property window shows the settings (label, font, color, size, etc.) of a control (such as a command button or text box) placed on the user form. For information on user forms, refer to User Form.

You can also set properties by programming in the code window.

To display the properties window, do one of the following:

1. On the **View** menu, click **Properties Window**.
2. Press **F4** key on the keyboard.
3. On the toolbar, click .

Closing Visual Basic Editor

This section describes how to quit Visual Basic Editor. Close the Visual Basic Editor using either one of the following methods:

- On Visual Basic Editor's **File** menu, click **Close and Return to E5052**.

- Within Visual Basic Editor, press **Alt** + **Q** keys on the keyboard.

- **Macro Setup** > **VBA Editor Menu** > **Close Editor** (E5052B measurement screen)

NOTE Whenever you launch Visual Basic Editor, it automatically displays the project files you were working with in the previous session. However, once you turn off the power to the E5052B, the project files kept in memory will be lost; therefore, it is strongly recommended to save your VBA programs before you turn off the power.

Other topics about Operation Basics

Switching to the E5052B Measurement Screen

You can switch to the E5052B measurement screen without closing Visual Basic Editor.

- On the **View** menu, click **E5052**.

- Press **Alt** + **F11** keys on the keyboard.

  - On the toolbar, click "E5052B" icon.

  - Press **Focus** key on the E5052B front panel.

Other topics about Operation Basics

Making a Preparation Before Coding

- A Project and Three Type of Modules

- Displaying a Code Window

Other topics about Operation Basics

A Project and Three Type of Modules

Project Explorer displays a list of files (modules) that are used in the E5052B VBA. This section describes a project composed of a number of files (modules) and three types of modules ("user form", "standard," and "class"). Each type of module serves its own purposes as described below.

Project

When you develop an application within the E5052B's VBA environment, you use a number of VBA program files (modules), and manage them as one project. The project is saved with the file extension ".vba".

User Form

A user form contains controls such as buttons and text boxes. You can code event-driven procedures that are invoked when a particular event occurs on a particular control, thereby creating a user interface. The user form is saved with the file extension ".frm".

Standard module

A standard module contains a collection of one or more procedures (subprograms enclosed between Sub and End Sub). One typical use of a standard module is to contain shared subroutines and globally called functions. The standard module is saved with the file extension ".bas".

Class Module

A class module contains both data and procedures and acts as one object. Once you have created a class module that serves as an object, you can create any number of instances of that object by naming each instance as an object variable. While each procedure must be unique in a standard module, you can have multiple instances of an object created through a class module. The class module is saved with the file extension ".cls".

Displaying a Code Window

The code windows appear on the Visual Basic Editor by inserting the modules in a project. You can do coding (programming) on this code windows practically.

The E5052B's VBA environment does not allow you to manage multiple projects. When the current project is existing in the Visual Basic Editor by loading the saved project file, you can replace the current project with a new project by the following method from the E5052B measurement screen.

1. **Macro Setup** > **VBA Editor Menu** > **New Project**

NOTE When you replace the current project with a new project, the message whether or not the current project is saved may appear. If you want to save the project, click **Yes** button to display a dialog box for saving. For saving the project, see Saving a Project.

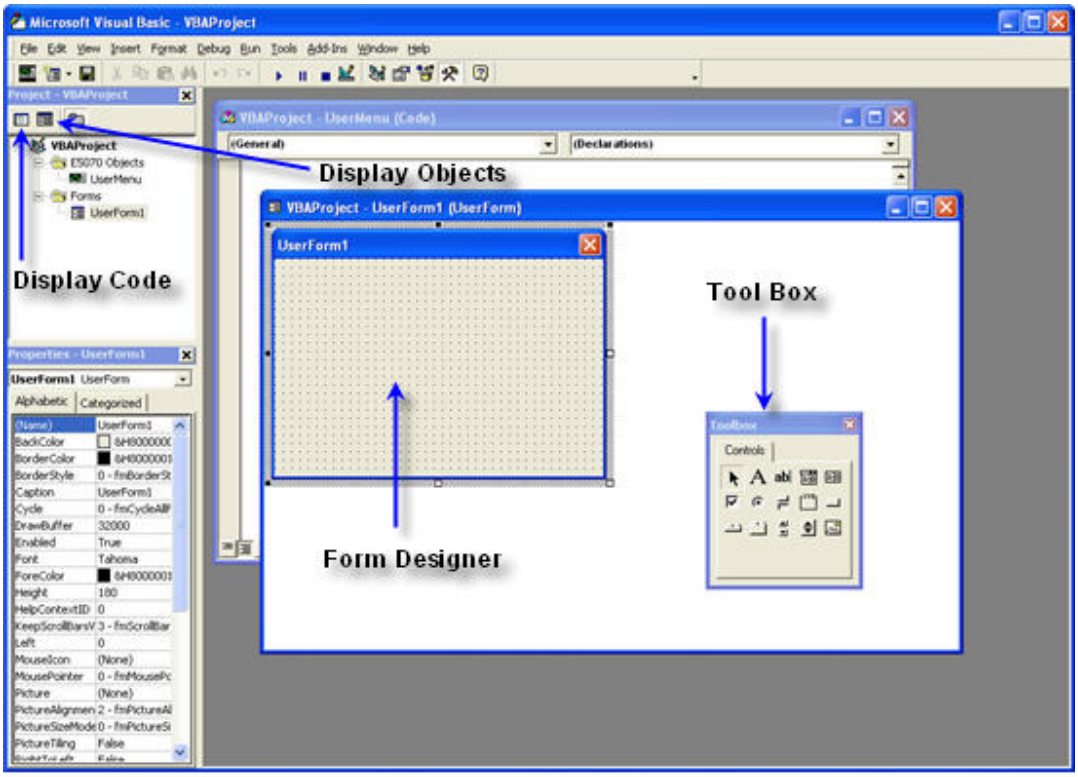
Inserting the User Form

Within Visual Basic Editor, do one of the following to add a user form to your project.

1. On the **Insert** menu, click **UserForm**.
2. On the toolbar, click "Insert User Form/Standard Module/Class Module/Procedure" icon, and click **UserForm**.
3. In Project Explorer, right-click the "VBAProject" icon, and click **Insert** > **UserForm**.

NOTE Adding a user form does not automatically open the code window for that user form. To open the code window, click the "Display Code" icon on Project Explorer in the following figure or double-click a control placed on the user form.

Adding a user form



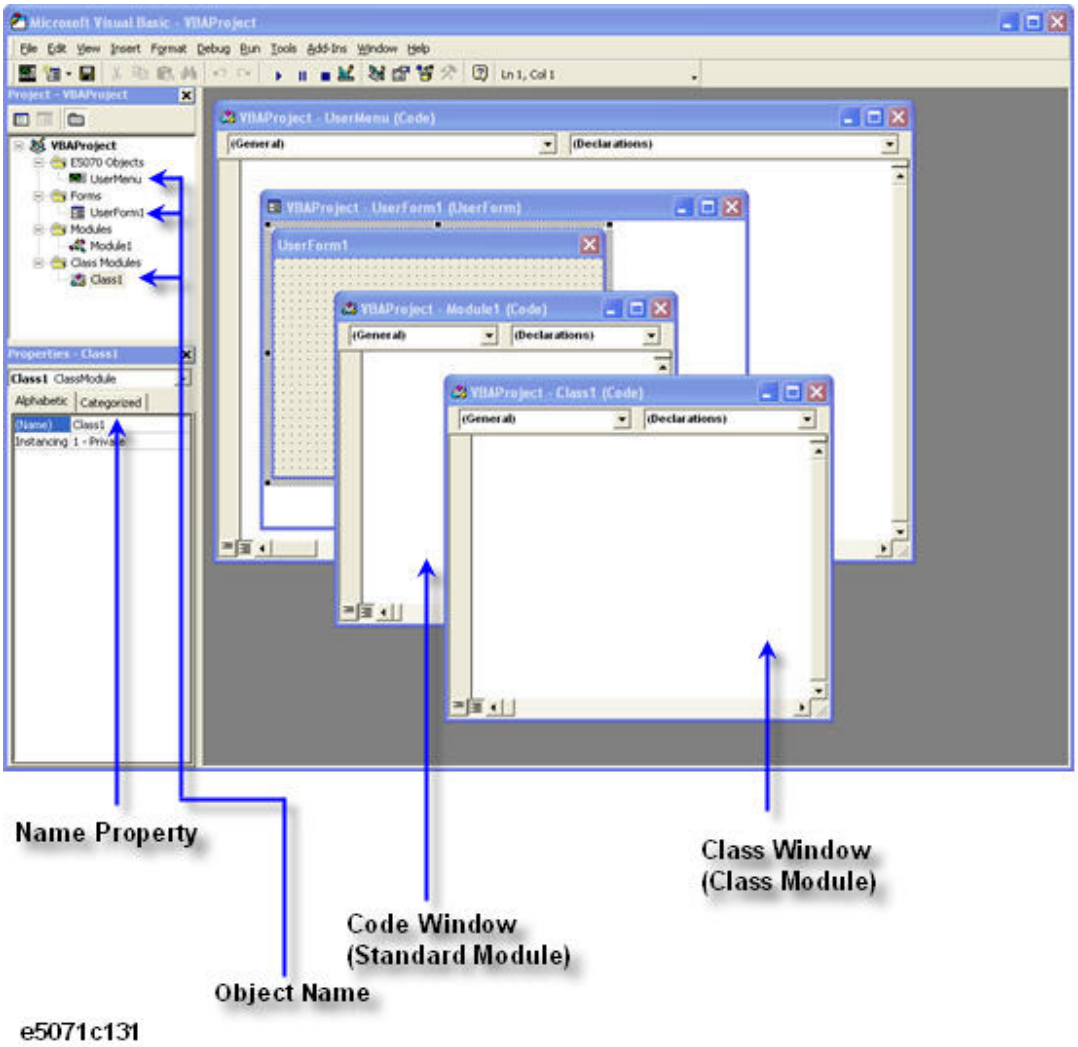
e5071c130

Inserting the Standard Module

Within Visual Basic Editor, do one of the following to add a standard module to your project.

1. On the **Insert** menu, click **Module**.
2. On the toolbar, click "Insert User Form/Standard Module/Class Module/Procedure" icon, and click **Module**.
3. In Project Explorer, right-click the "VBAProject" icon, and click **Insert** > **Module**.

Adding a standard module/class module



Inserting the Class Module

Within Visual Basic Editor, do one of the following to add a class module to your project.

1. On the **Insert** menu, click **ClassModule**.
2. On the toolbar, click "Insert User Form/Standard Module/Class Module/Procedure" icon, and click **ClassModule**.
3. In Project Explorer, right-click the "VBAProject" icon, and click **Insert** > **ClassModule**.

Deleting Modules

You can delete any unnecessary module from the project within Visual Basic Editor. The following procedure assumes that you want to delete a class module named "Class1".

1. In Project Explorer, click the "Class1" class module under the "Class Modules" icon to highlight it.
2. Delete the "Class1" class module using one of the following methods:
    a. On the **File** menu, click **Remove Class1...**.
    b. Click the right mouse button, and click **Remove Class1...**.
3. When you are prompted to confirm whether to export (save) "Class1", click **No**. Alternatively, you can click **Yes** if you want to save the module.

Coding a VBA Program

- Overview
- User Interface Elements of a Code Window
- Creating a Simple VBA Program
- Auto-complete Feature

Other topics about Operation Basics

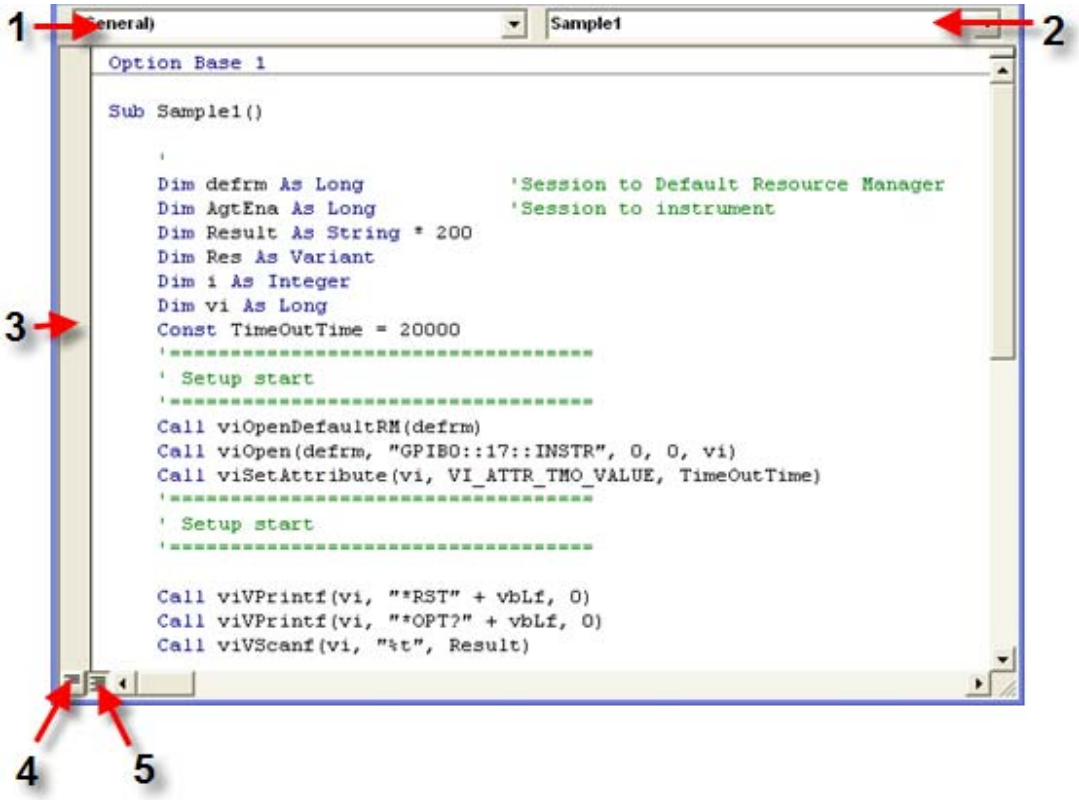
Overview

This section provides descriptive information on the user interface elements of a code window that lets you code a VBA program, and walks through a sample program (procedure) that finds the maximum value contained in an array so you can gain insight into how to create your own programs.

User Interface Elements of a Code Window

A code window is where you code a VBA program. When you are working with a user form, you can open the code window for that user form by double-clicking a control (such as a button or text box) placed on the form. Similarly, when you are working with a standard or class module, you can open the code window associated with that module by double-clicking the module's icon in Project Explorer.

Code window for a standard module



e5071c251

1. Object box

Provides a list of objects currently used within the code window.

2. Procedure box

Provides a list of procedures that reside within the code window. When you are working with a user form, this provides a list of events (actions such as click or double-click).

3. Margin indicator bar

Primarily intended for use when debugging a program.

4. Show Procedure button

Displays only the procedure at the cursor position.

5. Show Module button

Displays the entire program contained in the code window.

Creating a Simple VBA Program

This section walks through a sample program that finds the maximum value contained in an array while breaking down the code into a number of blocks and describing what they do. Line numbers are added for description purpose only, and do not appear in the actual program source code.

Sample program that finds the maximum value contained in an array

```
10| Option Explicit
20|
30| Sub Maximum()
40|
50| Dim q As Variant
60| Dim x(100) As Integer
70| Dim i As Integer, n As Integer
80| Dim Start As Integer, Last As Integer, Num As Integer
90| Dim Maximum As Integer
100|
110| ' Defining the array
120| q = Array(7, -2, 3, -20, 15, -6, 27, -12, 9, -5, 18, 23, _
130| 9, -16, 22, 0)
140|
150| Start = LBound(q)
160| Last = UBound(q)
170| Num = Last - Start + 1
180|
190| For i = Start To Last
200| x(i) = q(i)
210| Next i
220|
230| Maximum = x(Start)
240|
250| For n = Start + 1 To Last
260| If x(n) > Maximum Then Maximum = x(n)
```

```
270| Next n
280|
290| MsgBox Maximum
300|
310| End Sub
```

Let us break down the code into a number of blocks and see what they do.

Line 10

This instruction mandates explicit declaration of variables.

Lines 30 to 310

The code enclosed between Sub Maximum() and End Sub will be executed within the E5052B's macro environment. Thus enclosed code is called a procedure. In this example, "Maximum" is the procedure name.

Lines 50 to 90

These lines declare data types of variables using Dim statements. A statement is the minimum instruction unit based on the syntax. The sample program declares the variable "q" as Variant, and the variables "x(100)", "i", "n", "Start", "Last", "Num", and "Maximum" as Integer. For a complete list of statements and data types supported by VBA, see VBA Online Help.

Line 110

Any text preceded by a comment indicator (') is treated as a comment.

Lines 120 to 130

These lines use VBA's Array function to initialize the array. The q() array contains elements delimited with commas in the ascending order of index numbers (zero-based). A combination of a space and underscore (_) is used to continue the statement across two or more lines.

Line 150

Stores the starting index number of the q array into the Start variable.

Line 160

Stores the last index number of the q array into the Last variable.

Line 170

Stores the number of elements in the q array into the Num variable.

Lines 190 to 210 and Lines 250 to 270

The code within each For ...Next statement is iterated until the counter reaches the specific number.

Line 200

Stores the contents of the q array (Variant) into the x variable (Integer).

Line 230

Uses the first element of the x array as the tentative maximum value.

Line 260

Compares the tentative maximum value with each of elements that follow; if an element is larger than the tentative maximum value, then that element is used as the tentative maximum value.

Line 290

Uses a message box function to display the maximum value. For a complete list of functions supported by VBA, see VBA Online Help.

NOTE The above sample program consists of a single procedure contained in a single module. However, when you deal with procedures and variables across multiple modules, you should be aware of the scope of variables and procedures.

Auto-complete Feature

When you use COM objects in Visual Basic Editor, the editor's auto-complete feature allows you to easily type in keywords without misspelling them.

The following procedure assumes that you are entering the SCPI.INITiate.FP.CONTinuous object.

1. In a standard module, type **sub main** and press the **Enter** key. **End Sub** is automatically added.
2. Typing **scpi** followed by a dot (**.**) brings up a list of classes under the SCPI class.
3. Typing **in** automatically moves focus to **INITiate** in the list box.
4. Typing **INITiate** followed by a dot (.) brings up a list of classes under the SCPI class
5. Typing **f** automatically moves the focus to **FP** in the list box.

6. Typing **FP** followed by a dot (.) brings up a list of classes under the SCPI class.
7. Typing **c** automatically moves focus to **CONTinuous** in the list box.
8. Typing **=** brings up a list box for setting a Boolean value (**True/False**).
9. Typing **t** automatically moves focus to **True**.
10. Pressing the **Enter** key completes the statement:
    SCPI.INITiate.FP.CONTinuous = True.

Saving a VBA program

- Overview
- Saving a Project
- Saving a Module (Exporting)

Other topics about Operation Basics

Overview

You can save VBA programs either as one complete project or on a module by module basis.

Saving a Project

When you opt to save your program as one complete project, you can have the files (modules) making up the project into a single package. A project is saved as a .vba file. You can save your program to a project file using one of the following two methods:

Saving a Project from Visual Basic Editor

1. Open the Save As dialog box by doing one of the following:
    - On the **File** menu, click **Save xxx.VBA**. "**xxx**" represents the file name.
    - On the toolbar, click "Save Project File" icon.
    - Press **Ctrl** + **S** keys on the keyboard.
2. The Save As dialog box appears. Specify the file name and location (drive or folder) and click **Save**.

E5052B Saving a Project from the E5052B Measurement Screen

1. Display the E5052B measurement screen following the instructions given in Switching to the E5052B Measurement Screen.
2. Open the Save As dialog box using the following key sequence:
   - **Macro Setup** > **VBA Editor Menu** > **Save Project**
3. The **Save As dialog box** appears. Specify the file name and location (drive or folder) and click Save.

Saving a Module (Exporting)

Alternatively, you can save each module (user form, standard, or class) of your VBA program individually. To save a module, you must use Visual Basic Editor. User forms are saved as .frm files, standard modules as .bas files, and class modules as .cls files.

a. In Project Explorer, click the file name that appears under the desired module icon to highlight it.

b. Open the Export File dialog box by doing one of the following:
   - On the **File** menu, click **Export File...**.
   - Click the right mouse button, and click **Export File...**.
   - Press **Ctrl** + **E** keys on the keyboard.

c. The **Export File dialog box** appears. Specify the file name and location (drive or folder) and click **Save**.

Loading a VBA Program

- Overview
- Loading a Project

Other topics about Operation Basics

Overview

Once you have saved a project or module file, you can load it later whenever necessary.

Loading a Project

You can load a saved project file either from the E5052B measurement screen or by specifying that the project file be automatically loaded when the power is turned on.

Loading a Project from the E5052B Measurement Screen

1. Select **Macro Setup** > **VBA Editor Menu** > **Load Project**.

   NOTE When the another project has already been loaded on the Visual Basic Editor, the message whether or not the current project is saved may appear. If you want to save the project, click **Yes** button to display a dialog box for saving. For saving the project, see Saving a Project.

2. The Open dialog box appears. Specify the file name and location (drive or folder) of the file you want to load and click **Open**.

The Open dialog box has the following user interface elements:

1. **Look in**: Specify the location (drive or folder) where the project resides.
2. **File name**: Specify the file name of the project you want to load.
3. **Files of type**: Select the type of the file you want load. Normally, you should select **VBA Project Files [*.vba]**.
4. **Open**: Clicking this button loads the project.
5. **Cancel**: Clicking this button closes the Open dialog box and brings you back to the main screen.

Automatically Loading a Project at Power-On

Once you have saved a project file that satisfies the following conditions, the project will be automatically loaded whenever the power is turned ON.

Auto-loaded project	Conditions
Directory where the project resides.	**C:** or **F:**
Project file name	autoload.vba

If there is the file named "autoload.vba" in both the A drive and the D drive, the file in the A drive is used.

Loading a Module (Importing)

To load a saved module into a project, you must use Visual Basic Editor.

1. In Project Explorer, click the file name that appears under the desired module icon to highlight it.

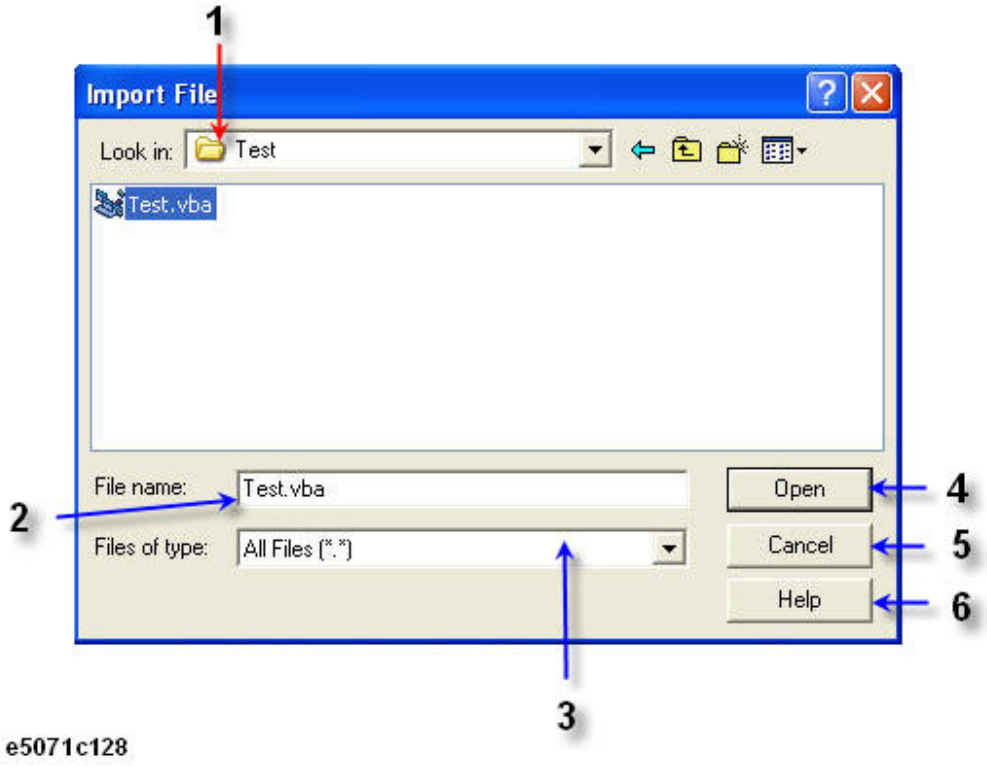
2. Open the Import File dialog box by doing one of the following:

   - On the **File** menu, click **Import File...**.
   - In Project Explorer, right-click the "VBAProject" icon, and click **Import File...**.
   - Press **Ctrl** + **M** keys on the keyboard.

3. The Import File dialog box appears. Specify the file name and location (drive or folder) of the file (module) you want to load and click **Open**.

4. The Import File dialog box has the following user interface elements:

Import File dialog box



The Import File dialog box has the following user interface elements:

1. **Look in**: Specify the location (drive or folder) where the module resides.

2. **File name**: Specify the file name of the module you want to load.
3. **Files of type**: Select the type of the file you want load. Normally, you should select **VB Files [*.frm,*.bas,*.cls]**.
4. **Open**: Clicking this button loads the module.
5. **Cancel**: Clicking this button closes the Import File dialog box and brings you back to the main screen.
6. **Help**: Clicking this button brings up VBA Online Help.

Running a VBA Program

- Overview
- Running a Previously Loaded VBA Program
- Loading and Executing Programs in Batch Process

Other topics about Operation Basics

Overview

The E5052B provides 2 methods to execute a VBA program: executing a program that you previously loaded and loading and executing a program in a batch process.

Running a Previously Loaded VBA Program

Running a Program from Visual Basic Editor

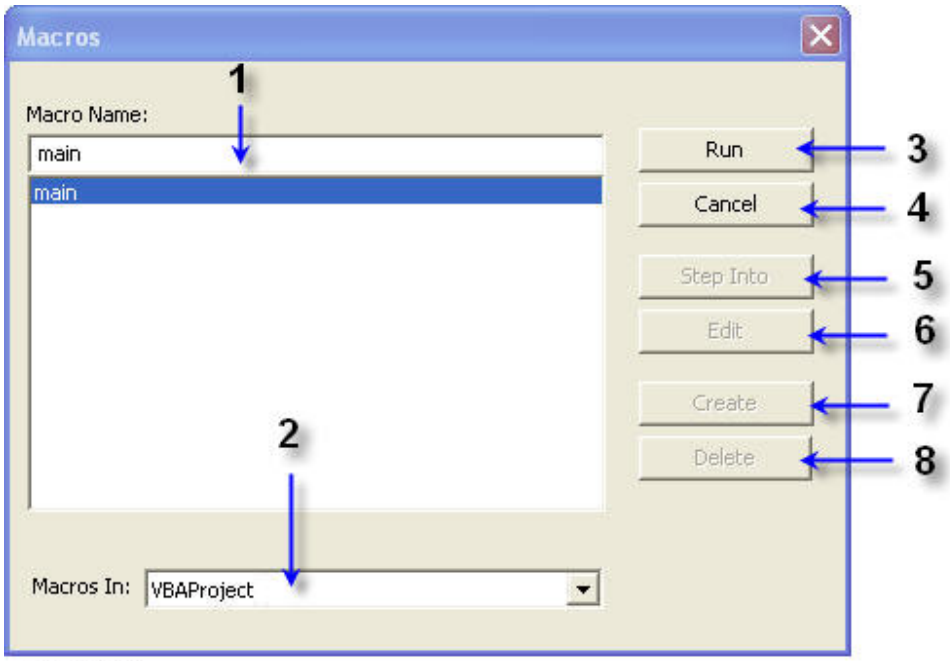
The E5052B allows you to run a previous loaded VBA program using one of the four methods listed below.

1. Open the Macros dialog box by doing either one of the following:
    - On the **Run** menu, click **Run Macro**.
    - On the **Tools** menu, click **Macros...**.
    - On the toolbar, click "Run Macro" icon.
    - Press **F5** key on the keyboard.

NOTE Doing the above steps with the cursor positioned within a procedure in the code window immediately runs the program without displaying the Macros dialog box.

2. In the Macros dialog box, select the VBA program (procedure name) you want to run, and click the **Run** button.

Macros dialog box



e5071c133

The Macros dialog box has the following user interface elements:

1. **Macro Name**: Select the VBA program (procedure name) you want to run from the list box so its name appears here.

2. **Macro In**: Specify the project that contains the VBA program you want to run. Normally, use the default.

3. **Run**: Clicking this button runs the selected VBA program (procedure).

4. **Cancel**: Clicking this button closes the Macros dialog box and brings you back to the main screen.

5. **Step Into**: Clicking this button brings up Visual Basic Editor and put it into step-in mode, where the selected VBA program is run step by step. This mode is primarily intended for use when debugging a VBA program. For more information on step-in mode, see Debug Toolbar.

6. **Edit**: Displays the code of the selected VBA program. You can use this for re-editing your code.

7. **Create**: This button is normally dimmed.

8. **Delete**: Clicking this button deletes the selected VBA program. Take care not to inadvertently delete your VBA program before saving it.

The Macros dialog provides access to subprograms (procedures enclosed between Sub and End Sub) created in a standard module.

Running a Program from the E5052B Measurement Screen

The E5052B allows you to run a program from E5052B screen using one of the four methods listed below.

1. Display the E5052B measurement screen following the instructions given in Switching to the E5052B Measurement Screen.

2. Run the VBA program (procedure) using the following key sequence:

- **Macro Setup** > **Select Macro - Module xxx**
  where "**Module**" is the object name (Name property shown in the property window) and "**xxx**" is the procedure name.

- Press the **Macro Run** key on the E5052B front panel. For a program to be run from the measurement screen, its procedure name must be "Main" (subprogram enclosed between Sub Main() and End Sub), and its object name (Name property as displayed in the property window) must be "Module1".

  NOTE When you are working with the E5052B measurement screen, the E5052B's macro environment only provides access to those VBA programs that are created as subprograms (enclosed between Sub and End Sub) in a standard module.

Loading and Executing Programs in Batch Process

This section describes how to load and execute a program (VBA project) in a batch process by pressing the softkey corresponding to the program name.

1. Save the VBA program (VBA project file) into the following folder.
   D:\VBA

   NOTE This feature is available only for programs saved in **F:\VBA**. This feature is not available for programs saved in subfolders of **F:\VBA**.

NOTE When copying a VBA program to F:\VBA from another folder, copy all the files necessary to execute the program to appropriate folders. When copying a factory-installed VBA program into F:\VBA, choose only its VBA project file.

2. Press **Macro Seup** key.
3. Click **Load & Run**.
4. Press the softkey corresponding to the VBA project file name of the program you want to execute. The pressed VBA project is loaded and the program whose procedure name is set to "Main" (subprogram enclosed between Sub Main() and End Sub) and whose object name (Name property as displayed in the property window) is set to "**Module1**" is executed.

NOTE There is no limit to the number of VBA project files that can be saved in **F:\VBA**. However, the maximum number of programs that can be displayed as softkeys is 50.

- File names of the VBA projects saved in **F:\VBA** are displayed as softkeys in alphabetical order.
- The maximum number of characters that can be displayed in a softkey is 12. If a file name has 13 or more characters, "..." is added to the 12th character from the beginning of the program name and displayed. In this case a .vba extension is omitted.

Stopping a VBA Program

- Stopping a Procedure
- Abruptly Terminating a VBA Program

Other topics about Operation Basics

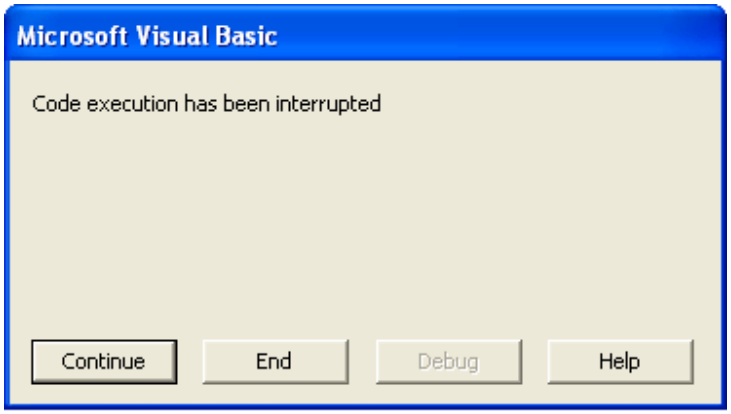
Stopping a Procedure

This section describes how to break a procedure during the execution of a VBA program.

1. To break the running VBA program, do one of the following:
    - On the **Run** menu, click **Break**.
    - On the toolbar, click "Break Macro" icon.

- Press **Ctrl** + **Break** keys on the keyboard.
- **Macro Setup** > **Stop** (E5052B measurement screen)
- Press **Macro Break** key on the E5052B front panel.

2. A dialog box is displayed through forced interrupts, and the program is suspended.



e5071c250

Select one of the following:

- **Continue**: Resumes the execution of the program.
- **End**: Terminates the VBA program.
- **Debug**: Displays a run-time error.
- **Help**: Brings up VBA Online Help.

Abruptly Terminating a VBA Program

This section describes how to abruptly terminate a running procedure. When abruptly terminating the VBA program by the below methods, the "Program interrupted" message is shown in the instrument status bar on the bottom of the LCD display.

To terminate the running VBA program, do one of the following:

- On the **Run** menu, click **Reset**.
- On the toolbar, click "Reset Macro" icon.
- Insert an *End* statement into your code.

Errors and Debugging

- Type of Errors
- Using a Debug Tool

Other topics about Operation Basics

Type of Errors

Errors in VBA programs are classified into the following two types:

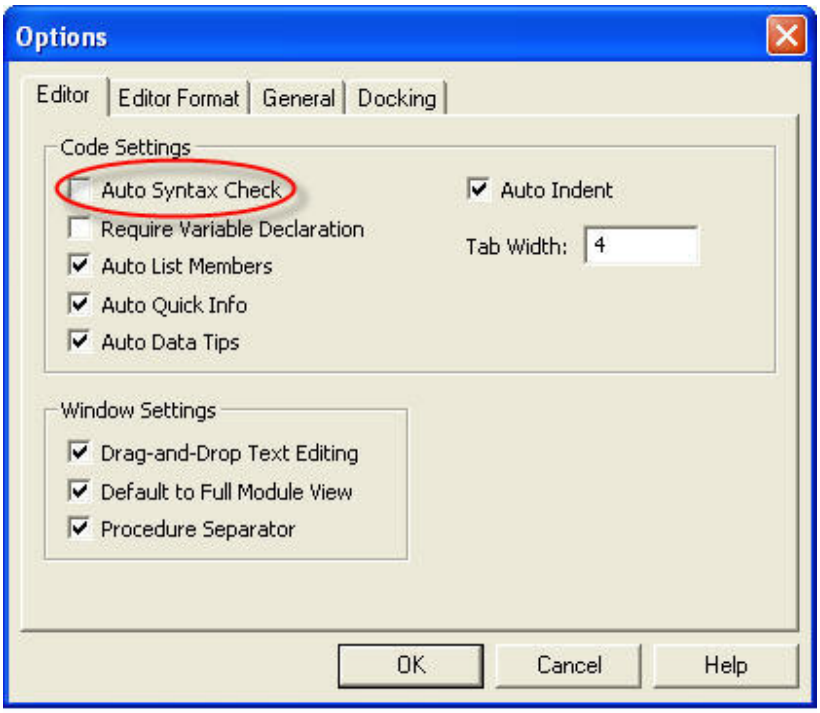
Syntax errors

A syntax error is generated when Visual Basic Editor detects an invalid statement that violates the Visual Basic syntax rules. For example, misspelled keywords generate syntax errors. An error dialog box appears that indicates the error message, and highlight the invalid statement in red. To get detailed information on the error, click the **HELP** button in the error dialog box to display the help topic on the error. You cannot run the macro until you correct the syntax error.

The E5052B VBA environment is by default configured to automatically check for syntax errors, but you can disable the auto syntax check feature using the following steps:

1. On the **Tools** menu, click **Options...**.

2. On the **Editor** tab, clear the **Auto Syntax Check** check box.



e5071c121

3. Click the **OK** button.

Run-time Errors

A run-time error is generated when a VBA program attempts to execute an invalid statement at run time. When a run-time error is generated, the program is stopped at the invalid statement, and an error dialog box appears. You can terminate the program by clicking the END button in the error dialog box. Also, you can click the DEBUG button in the error dialog box to identify the statement that caused the error. In this case, the statement in question is highlighted in yellow.

NOTE Some run-time errors occur under particular conditions, even though a program run without occurring the errors under normal conditions. For example, the "Target value not found" error that occurs when a program that analyzes the results using the Marker Bandwidth Search feature fail to perform bandwidth search because the marker is not in the appropriate position, the "Ecal module not in RF path" error that occurs when a program that performs calibrations using a ECal module fail to measure the calibration data because the ECal module is not appropriately connected to test ports, and so on.

Using a Debug Tool

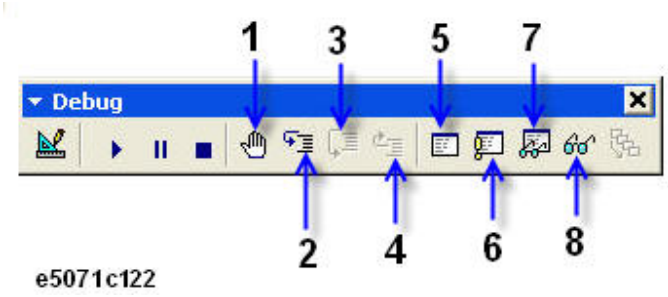
The E5052B's VBA environment provides a variety of debug tools that help you identify logical errors. Detailed information on using the debug tools is covered in VBA Online Help and books on VBA.

Debug Toolbar

The debug toolbar provides tool buttons that allow you to easily access various debug tools.

On the **View** menu, click **Toolbars** > **Debug**.

Debug toolbar



1. Set/clear break points (keyboard **F9**): Puts a break point at the cursor position or clears an existing break point.
2. Step-in (keyboard: **F8**): Runs the VBA program step by step. If the current program contains a call to another procedure, that procedure is also run step by step.
3. Step-over (keyboard: **Shift** + **F8**): Runs the VBA program step by step. If the current program contains a call to another procedure, that procedure is run as one line.
4. Step-out (keyboard: **Ctrl** + **Shift** + **F8**): Executes the remaining lines of the function where the execution point is currently placed.
5. Local window: Opens the local window that shows the current values of local variables.
6. Immediate window (keyboard: **Ctrl** + **G**): Opens the immediate window that evaluates entered values of variables or expressions.
7. Watch window: Opens the watch window that displays the current value of a specified expression.

8. Quick window (keyboard: **Shift** + **F9**): Displays the current value of a specified expression in a dialog box.

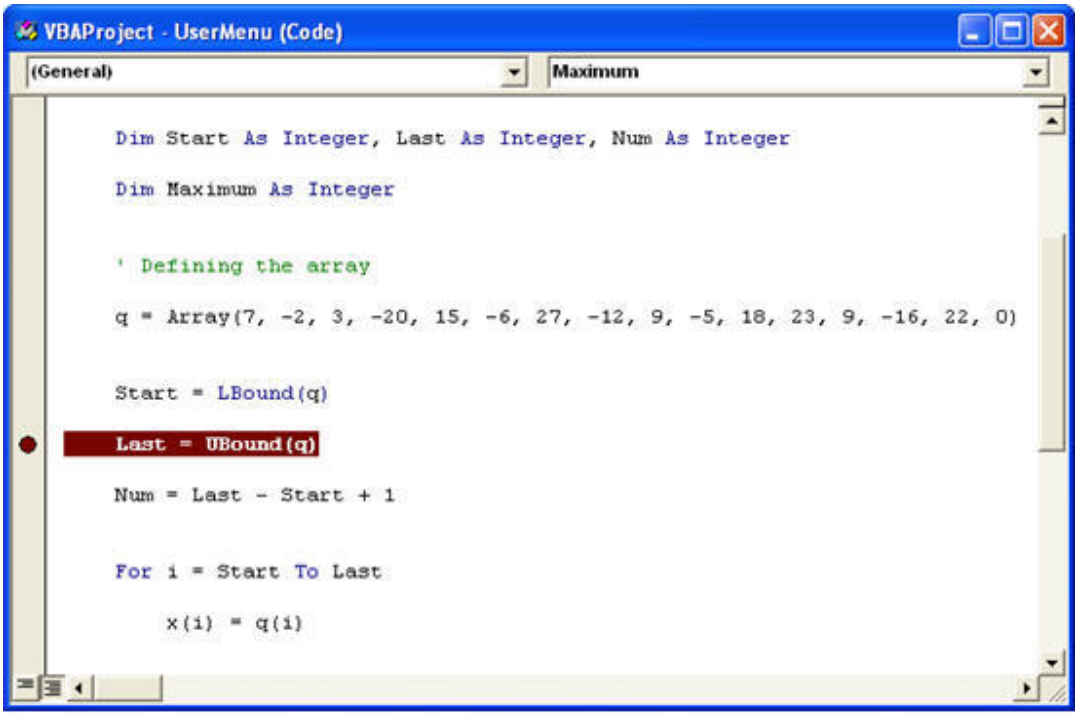
Setting a Break Point

By placing a break point at a particular statement in a VBA program, you can automatically suspend the program when it is executed to that statement.

When you put a break point at a line, the line is highlighted in umber as shown in the following figure. To set a break point, do one of the following:

- Place the cursor at the desired line of code, and click the "Set/clear break points" button on the debug toolbar.
- Click anywhere in the margin indicator bar of the code window.

Setting a break point



e5071c123

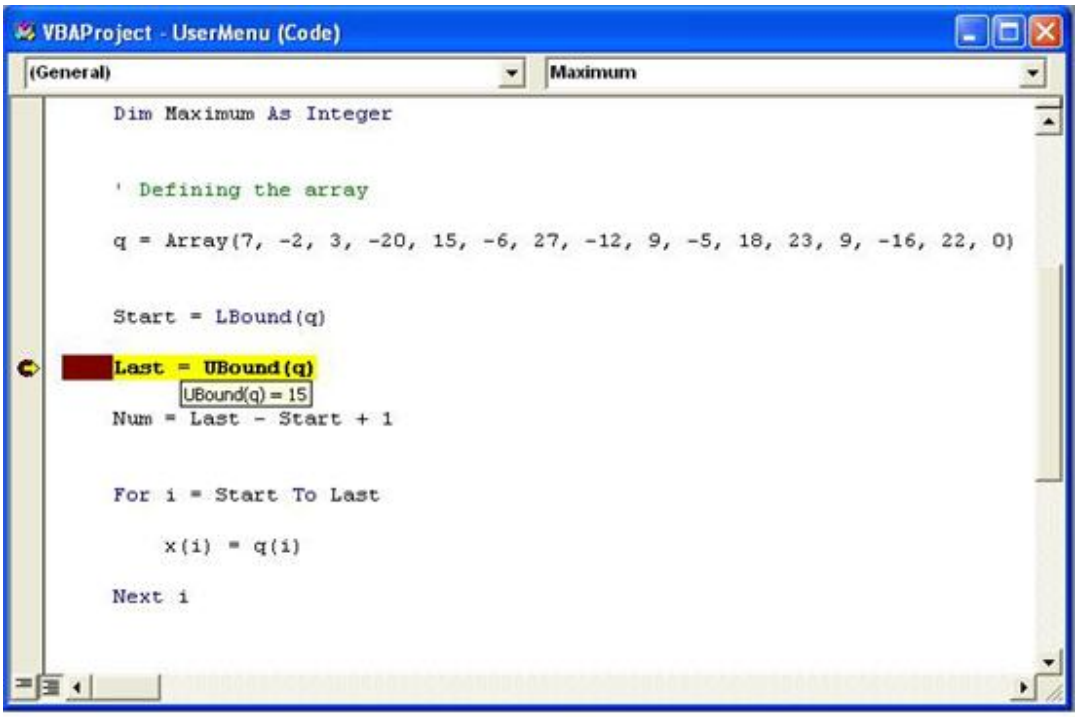
Monitoring Variable or Property Values

With your VBA program suspended, you can use the following debug tool to monitor variables or properties. To do this, you must set a break point, run the VBA program, and suspend it.

Data Hint

When you point to the variable or expression of interest, Data Hint shows the current value as shown in the following figure.

Data Hint



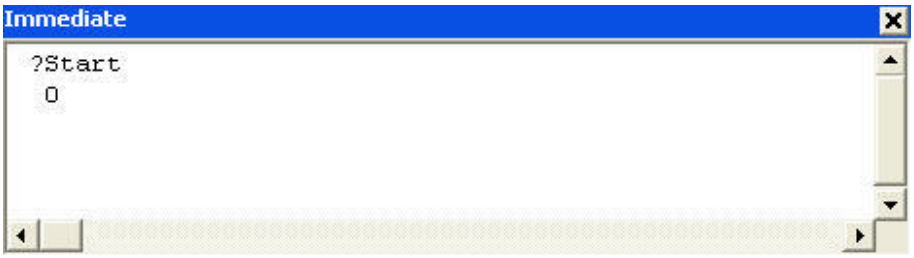
e5071c127

Immediate Window

To display the immediate window, click  on the debug toolbar.

In the immediate window, enter a question mark (?) followed by the variable or expression whose value you want to check, and press the Enter key, as shown in the following figure. The current value appears in the line that follows.

Immediate window

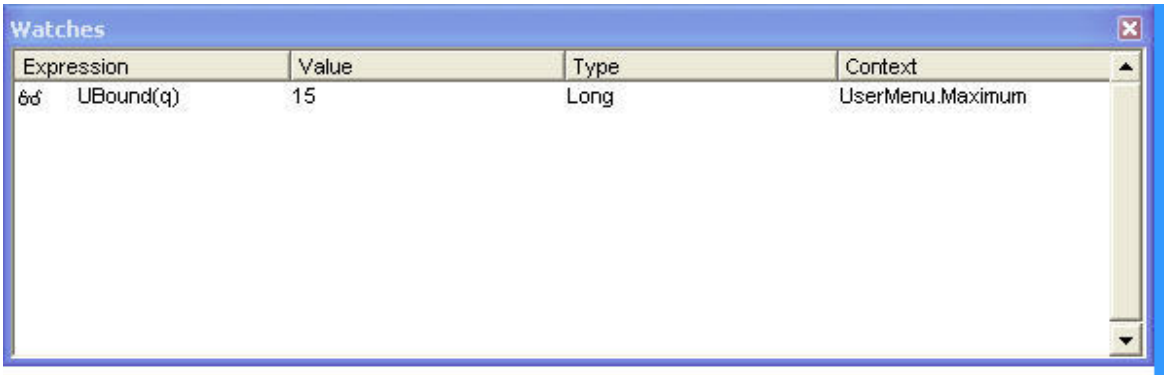


e5071c124

Watch Window

To display the watch window, click the "Watch Window" button on the debug toolbar.

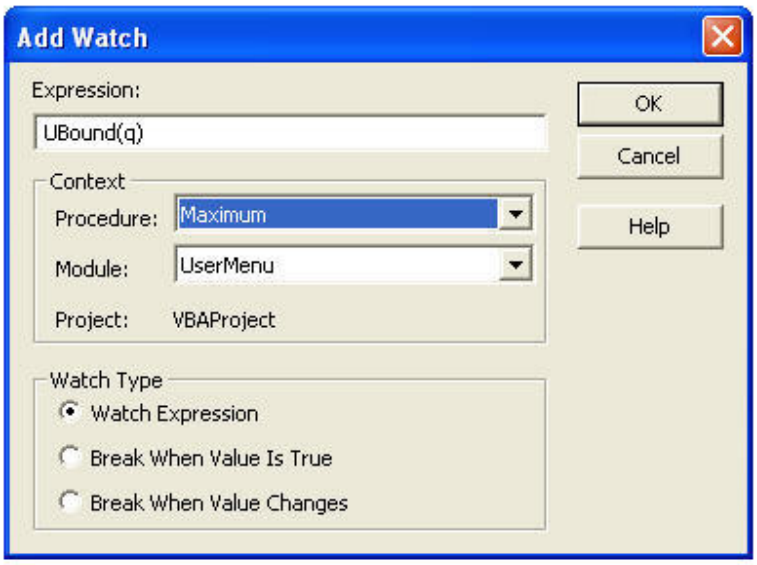
Watch window



e5071c126

1. On the **Debug** menu, click **Add Watch...**. to open the Add Watch dialog box.
2. As shown in the following figure, you can specify an expression of interest as a watch expression to always monitor its value.
3. Click the **OK** button.

Add Watch dialog box



e5071c125

Quick Watch

In the code window, select a variable or expression whose value you want to watch. On the debug toolbar, click the "Quick Watch" button to open the Quick Watch dialog box. The dialog box displays the current value of your specified variable or expression.

Also, you can click the **Add** button in the Quick Watch dialog box to specify the current expression as a watch expression.

Printing Output Values in the Echo Window

- Overview
- Entering Values Output to Echo Window
- Opening Echo Window
- Clearing Values Output from Echo Window
- Echo Font Size

Other topics about Operation Basics

Overview

The echo window, which appears in the lower part of the E5052B measurement screen, can be used to display a message or the return value (data) of an object.

Entering Values Output to Echo Window

You can use the COM objects listed below to enter values output to the echo window.

- SCPI.DISPlay.ECHO.DATA

Opening Echo Window

You can use the COM objects listed below to open the echo window.

- SCPI.DISPlay.TABLe.STATe

Alternatively, you can also open the echo window using the following key sequence:

Macro Setup > **Echo Window Menu** > **Echo Window (ON)**

Clearing Values Output from Echo Window

You can use the COM object shown below to clear values output from the echo window.

- SCPI.DISPlay.ECHO.CLEar

Alternatively, you can also clear values output from the echo window using the following key sequence:

Macro Setup > **Echo Window Menu** > **Clear Echo**

Echo Font Size

You can use the COM object shown below to set the font size of the echo window

- SCPI.DISPlay.ECHO.FSIZe

Alternatively, you can also set the font size using the following key sequence:

Macro Setup > **Echo Window Menu** > **Echo Font Size**

Using Advanced Techniques

- Accessing a List of E5052B COM Objects
- Using Automatic Library References

Other topics about Operation Basics

Accessing a List of E5052B COM Objects

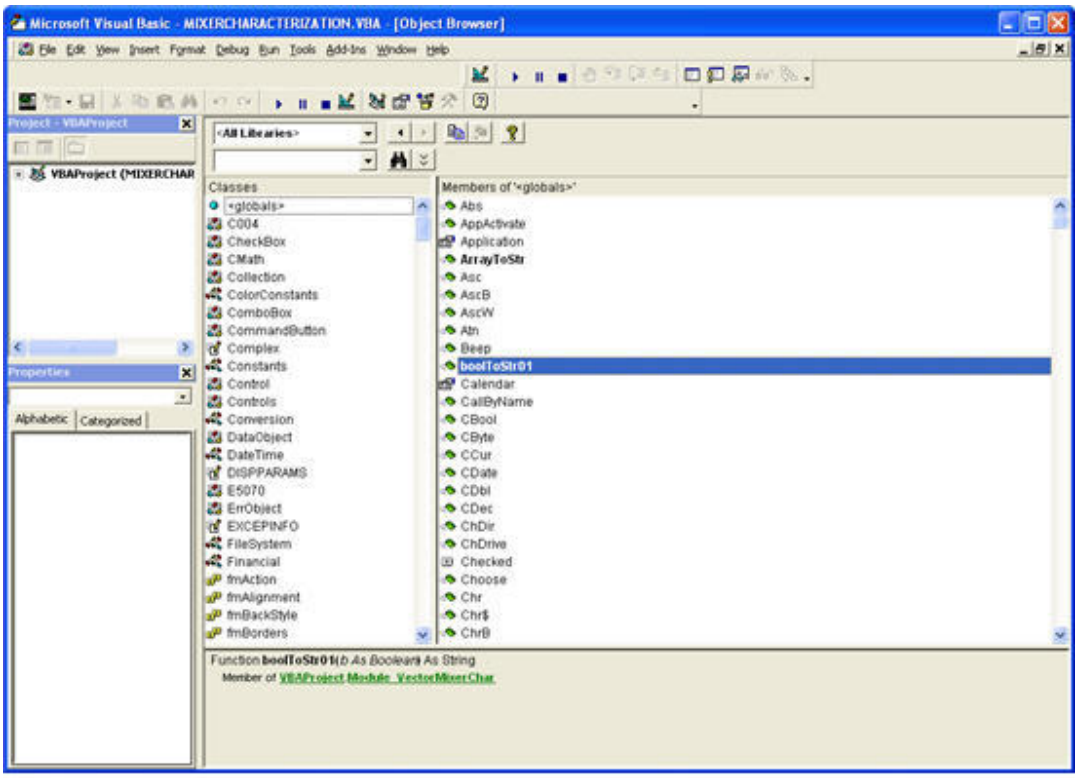
The E5052B VBA environment provides COM objects that support controlling the E5052B. When you are developing a program using E5052B COM objects, you can access a list of E5052B COM objects by opening Object Browser within Visual Basic Editor.

1. To open Object Browser, do one of the following:
   - On the **View** menu, click **Object Browser**.
   - On the toolbar, click "Object Browser" icon.
2. Select **E5052Lib** from the Project/Library box to display the E5052B library as shown in the following figure.

NOTE There are some COM objects NOT used in controlling with E5052B VBA in the list of the E5052B COM objects displayed on the

Object Browser. The COM objects NOT used in controlling with E5052B VBA are not described in the COM object reference.

How to use Object Browser



e5071c134

Using Automatic Library References

For libraries that satisfy the following conditions, the library reference will be automatically set whenever a new project is created and loaded (**Macro Setup** > **VBA Editor Menu** > **New Project**).

Automatically referenced libraries	Conditions
Directory where the library resides.	**D:\Agilent**
Extensions of libraries	olb, tlb, dll, or ocx

To check the library reference setting, you must use Visual Basic Editor.

Follow these steps to check the library reference setting.

- On the **Tools** menu, click **References...**.

NOTE A project sets the library reference when the project is created. Therefore, if the existing project is loaded, libraries added after the development of the project are not automatically set in the library reference.

Using VBA Online Help

- Overview
- Accessing VBA Online Help

Other topics about Operation Basics

Overview

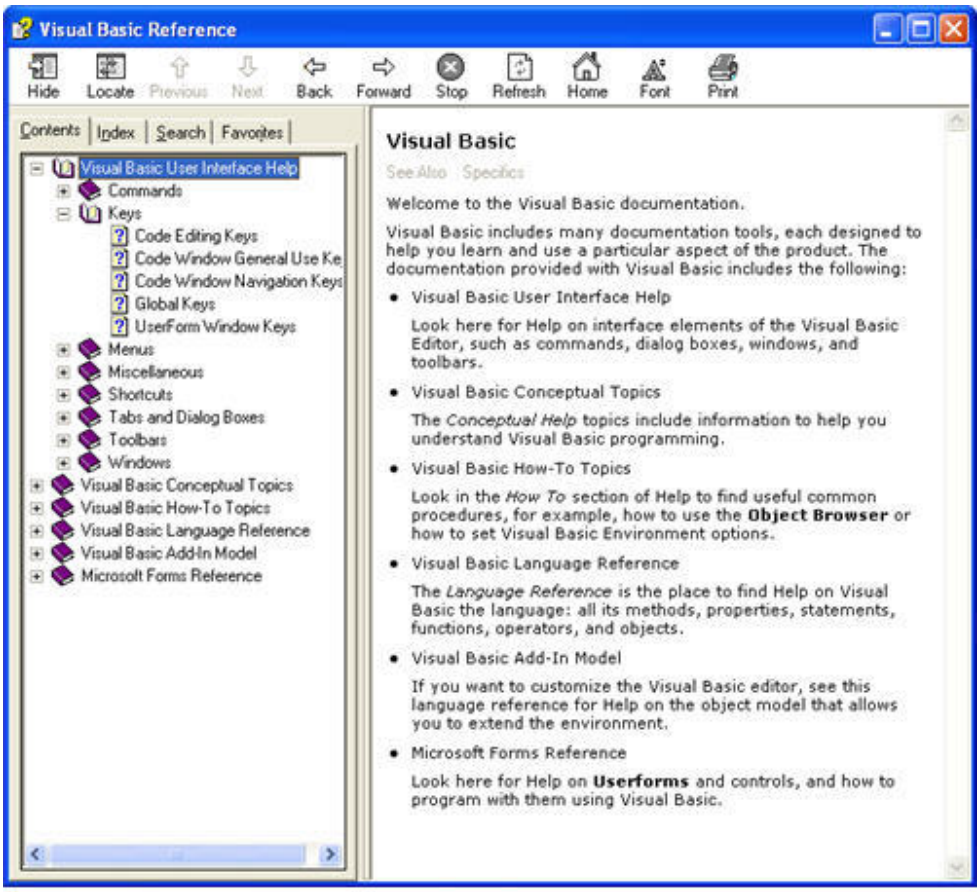
VBA Online Help provides useful topics, such as the VBA terminology or how to use a particular feature. In VBA Online Help, you can find a topic of interest through the Contents or by entering specific keywords.

Accessing VBA Online Help

From Visual Basic Editor, do one of the following to access the VBA Online Help screen.

- On the **Help** menu, click **Microsoft Visual Basic Help**.
- Press **F1** key on the keyboard.
- On the toolbar, click "VBA Help" icon.

VBA Online Help screen



e5071c135

Using the Contents Tab

Clicking the **Contents** tab in the VBA Online Help screen brings up the items listed below. The E5052B VBA Online Help has a hierarchical table of contents. Click an item to expand it, and then find a topic of interest.

- Visual Basic User Interface Help
- Visual Basic Conceptual Topics
- Visual Basic How-To Topics
- Visual Basic Language Reference
- Visual Basic Add-In Model
- Microsoft Forms Reference

When you need information on using Visual Basic Editor, use User Interface Help and How-To Topics as primary sources of information. Format of VBA program is covered in Visual Basic Conceptual Topics. Properties and methods supported by VBA are covered in Visual Basic Language Reference and Visual Basic Add-In Model. Information on using user forms is covered in Microsoft Forms Reference.

Using the Index Tab

In the VBA Online Help screen, click the **Index** tab, and enter a keyword(s) into the text box. For example, you may wish to search for "Sub" or "With" when you are writing your own code.

Looking up a Keyword in the Code within Visual Basic Editor

When you want to know the usage or meaning of a keyword contained in a sample program or some other code, you can quickly access the help topic on that keyword by moving the cursor to the keyword and pressing **F1** key.

Controlling the E5052B

- Controlling VBA Externally
- Detecting End of Measurement
- Executing a Procedure with a Softkey (user menu function)
- Limit Test
- Reading/Writing Measurement Data

Controlling VBA Externally

- Overview
- Executing VBA Using External Controller
- Receiving Termination of VBA Using External Controller
- Using User-defined Register
- Using User-defined Variables

Other topics about Controlling E5052B

Overview

This section describes how to control the E5052B's VBA externally.

Executing VBA Using External Controller

You can execute VBA from an external PC by running either macros or the user menu.

Running Macro

To start VBA from an external PC, use the following command:

- SCPI.MMEMory.LOAD.PROGram
- SCPI.PROGram.COM.EVENt
- SCPI.PROGram.SELected.STATe

Running User Menu

To execute the user menu from an external PC, use the following command:This command executes the first of the menu.

To execute the user menu, you must execute the user menu function in advance. For more information, refer to Running User Menu
and SCPI.PROGram.SKEY.Item(1).IMMediate

Receiving Termination of VBA Using External Controller

To allow the external controller to receive the termination of VBA from the E5052B, you may either confirm the operational status of VBA or use the user-defined register.

Confirming VBA's Operational Status

To confirm the operational status of VBA, use the following command:

- SCPI.PROGram.SELected.STATe

Using User-defined Register

To use the user-defined register, use the following command:For more information, refer to Using User-defined Register.

- SCPI.STATus.OPERation.BIT12.CLEar
- SCPI.STATus.OPERation.BIT12.CONDition
- SCPI.STATus.OPERation.BIT12.ENABle
- SCPI.STATus.OPERation.BIT12.EVENt
- SCPI.STATus.OPERation.BIT12.NTRansition
- SCPI.STATus.OPERation.BIT12.PTRansition
- SCPI.STATus.OPERation.BIT12.SET

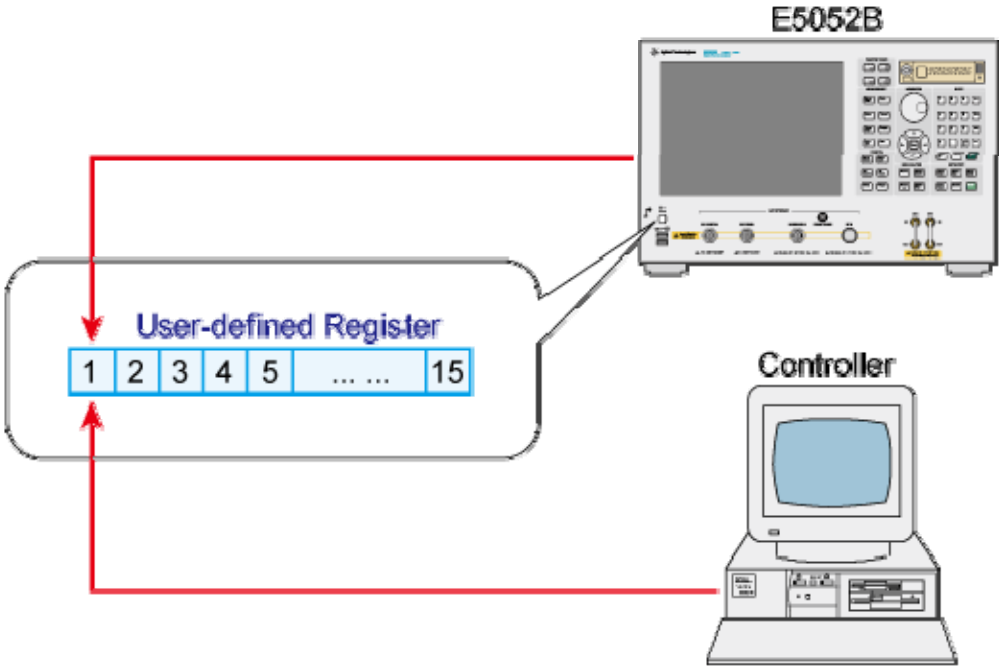
Using User-defined Register

The E5052B does not control the individual status flag of the user-defined register. In order to start the E5052B VBA using the external controller to obtain the end bit, the user must make the user-defined register to be controlled within the program.

The user can assign any register number to the user-defined register which he/she wants to use. Available register numbers are 1 to 15 (0 to 14 bit).

If you use the user-defined register, you must specify the same register number for both the external controller side and the E5052B's program side.

Reference of User-defined Register



ssa0076

The following is a sample program which:

1. Clears the first condition register to be used (bit 0).
2. These lines configure the instrument so that the operation use-defined status event register's bit 0 is set to 1, when the operation use-defined condition register's bit 0 is changed from 0 to 1 (i.e. positive transition).
3. Executes the user menu.
4. Repeats until the termination of VBA is detected.

A Sample Program to Control User-defined Register (On the Controller Side

Lines 30 to 90

An event occurs when the softkey is pressed from the user menu.

Lines 50 to 60

E5052B will be reset to the initial setting, when the first softkey is pressed.

Lines 70 to 80

The message box will appear, when the second softkey is pressed.

Lines 100 to 110

An event will occur when the triggered sweep ends.

Line 110

A message box will appear when the sweep ends.

Line 130

Sets a flag to the first of the user-defined register (bit 0).

Line 140

This procedure terminates the program.

A Sample Program to Control User-defined Register (On the E5052B Side)

```
10|Private Sub app_OnEvent(ByVal ReasonStr As String, ByVal OptionStr As String)
20| Select Case ReasonStr
30| Case "UserLabel"
40| Select Case OptionStr
50| Case 1
60| SCPI.SYSTem.PRESet
70| Case 2
80| MsgBox "Program ended"
90| End Select
100| Case "SweepEnd"
110| MsgBox "Sweep ended"
120| End Select
130| SCPI.STATus.OPERation.BIT12.SET = 0
140| End
150|End Sub
```

NOTE For more information on the user-defined register, refer to the status reporting system.

Using User-defined Variables

The E5052B has an area for which the users may set any value. The areas are divided for each data type.

An area can be used up to 10 (1 to 10) for each command.

The values set by a command cannot be removed by executing preset.

- SCPI.PROGram.VARiable.ARRay(1-10).DATA
- SCPI.PROGram.VARiable.ARRay(1-10).POINts
- SCPI.PROGram.VARiable.DOUBle(1-10)
- SCPI.PROGram.VARiable.INTeger(1-10)
- SCPI.PROGram.VARiable.STRing(1-10)

NOTE These commands do not refer to or change the results within the E5052B.

The following is a sample program for configuration, which demonstrates how to use the floating-point-formatted user-defined array.

Line 30 to 50

Define the variables.

Line 70

Assign values to the variables.

Line 90

Call the function to trigger the instrument.

Line 110

Configure marker X for frequency measurement (trace 1).

Line 120

Read out marker Y for frequency measurement (trace 1).

Line 130

Configure marker X for power measurement (trace 2).

Line 140

Read out marker Y for power measurement (trace 2).

Line 160

Set the value of marker Y for frequency measurement at the beginning of the user-defined array.

Line 170

Set the value of marker Y for power measurement to that of the 2nd dimension of the user-defined array.

Example of using floating point formatted user-defined array

```
10| Sub Main()
20|
```

```
30| Dim Vcc1 As Double
40| Dim FP_Freq As Double
50| Dim FP_Power As Double
60|
70| Vcc1 = 0.5
80|
90| Call FP_SingleSweep
100|
110| SCPI.CALCulate.FP.Trace(1).MARKer(1).X = Vcc1
120| FP_Freq = SCPI.CALCulate.FP.Trace(1).MARKer(1).Y
130| SCPI.CALCulate.FP.Trace(2).MARKer(1).X = Vcc1
140| FP_Power = SCPI.CALCulate.FP.Trace(2).MARKer(1).Y
150|
160| SCPI.PROGram.VARiable.DOUBle(1) = FP_Freq
170| SCPI.PROGram.VARiable.DOUBle(2) = FP_Power
180| End Sub
```

The following sample program demonstrates how to use the array-formatted user-defined array.

Line 30 to 50

Define the variables.

Line 70

Call the function to trigger the instrument.

Line 90

Read out the number of points and set it to the variable.

Line 100 to 110

Redefine the array (variable).

Line 130

Read out the trace data and set it to the variable.

Line 140

Read out the X-axis data and set them to the variable.

Line 160

Specify the number of data points at the beginning of the user-defined array.

Line 170

Specify the trace data at the beginning of the user-defined array.

Line 190

Specify the number of data points to that of the 2nd dimension of the user-defined array.

Line 200

Specify the X-axis data to that of the 2nd dimension of the user-defined array.

Example of using array formatted user-defined array

```
10| Sub Main()
20|
30| Dim PN_Nop As Long
40| Dim PN_Trace() As Double
50| Dim PN_Freq() As Double
60|
70| Call PN_SingleSweep
80|
90| PN_Nop = SCPI.SENSe.PN.SWEep.POINts
100| ReDim PN_Trace(PN_Nop - 1)
110| ReDim PN_Freq(PN_Nop - 1)
120|
130| PN_Trace = SCPI.CALCulate.PN.Trace.DATA.FDATa
140| PN_Freq = SCPI.CALCulate.PN.DATA.XDATa
150|
160| SCPI.PROGram.VARiable. Array (1).POINts = PN_Nop
170| SCPI.PROGram.VARiable.Array(1).DATA = PN_Trace
180|
190| SCPI.PROGram.VARiable.Array(2).POINts = PN_Nop
200| SCPI.PROGram.VARiable.Array(2).DATA = PN_Freq
210|
220| End Sub
```

Detecting End of Measurement

- Using the Status Register
- Using Event Interruption feature
- Interference between different interfaces

Other topics about Controlling E5052B

This section explains how to trigger the instrument to start a new measurement cycle and how to detect the end of a measurement cycle. For a detailed description of trigger detection, the trigger system, and the concept of triggering, refer to Making a Measurement.

You can detect the end of measurement by using either Using the Status Register or Using Event Interruption feature.

For the E5052B's VBA, regardless of the setting of the SCPI.TRIGger.SOPC command, neither SCPI.IEEE4882.OPC nor SCPI.IEEE4882.WAI can detect the end of measurement. Furthermore, the Parse command is not available.

Using the Status Register

The status of the E5052B can be monitored through the status register. For a complete description of the status report mechanism, including the specifications of each bit of the status register, refer to Status Reporting System.

If your program is based on SPCI commands, you can use SRQ (Service Request) interrupts to detect the end of measurement. For more information, refer to Waiting for the End of Measurement.

The following is a sample program that demonstrates how to use an SRQ to detect the end of measurement.

In this program, you can start a measurement cycle by pressing the command button on the user form (cmdExec), and a message box will appear when the measurement finishes.

Line 40

The trigger continuous mode is turned "OFF"

Lines 50 to 60

The trigger system switches to "Idle" state and clears the operation status event register.

Lines 70 to 80

These lines configure the instrument so that operation status event register's bit 4 is set to 1 only when the operation status condition register's bit 4 is changed from 1 to 0 (negative transition).

Lines 100 to 120

The trigger source is set to "Bus Trigger" to start a measurement cycle.

Lines 140 to 170

These lines repeat until the end of measurement is detected.

Line 150

The control is returned to Windows so that other applications may be executed.

Line 190

A message box appears when the end of measurement is detected.

Example of Using an Status Register to Detect the End of Measurement

```
10| Private Sub cmdExec_Click()
20| Dim i As Long
30|
40| SCPI.INITiate.SP.CONTinuous = False
50| SCPI.ABORt
60| SCPI.IEEE4882.CLS
70| SCPI.STATus.OPERation.PTRansition = 0
80| SCPI.STATus.OPERation.NTRansition = 16
90|
100| SCPI.TRIGger.SP.Source = "bus"
110| SCPI.INITiate.SP.CONTinuous = True
120| SCPI.IEEE4882.TRG
130|
140| Do While i = 0
150| DoEvents
160| i = SCPI.STATus.OPERation.EVENt
170| Loop
180|
190| MsgBox "end"
```

```
200|
210| End Sub
```

The E5052B's VBA program is executed as a part of E5052B's applications. Therefore, if any executed VBA program takes a long time before returning control to Windows, all processing operations other than VBA are suspended during this time. To prevent this, use DoEvents to return the control to E5052B applications.

Using Event Interruption feature

The E5052B provides an event interruption feature to allow you to detect the end of measurement. By enabling the E5052B Event softkey, you can obtain an end of sweep event.

For details, refer to the section on Executing a Procedure with a Softkey (user menu function).

NOTE To obtain an end of sweep event, you must create and execute a VBA program.

Interference between different interfaces

For the E5052B connected with two different interfaces (except for the VBA's COM interface), for example, GPIB and LAN, while a controller is waiting for the end of measurement with *OPC? command or *WAI command through GPIB interface, LAN interface is unavailable until the measurement completes.

On the other hand, for the E5052B connected with two different interfaces, for example, GPIB and VBA, VBA's COM interface is available, and commands can be executed even while a controller is waiting for the end of measurement with *OPC? command or *WAI command through GPIB interface. When the Parse command is executed, however, an error occurs.

Executing a Procedure with a Softkey (user menu function)

- Preparing to use the User Menu Function
- How to use the User Menu Function
- Simple usage example
- Argument for event occurrence

Other topics about Controlling E5052B

The E5052B lets you perform procedures assigned to specific softkeys (**Macro Setup** > **User Menu** > **User Label <1/2/3/4/5/6/7/8>**) without using user forms for the event activated by pressing the softkey. This function is called the user menu function.

NOTE You must create and execute a VBA program when using the user menu function.

Preparing to use the User Menu Function

Before using the user menu function, perform the following preparation.

Settings for Softkey Label Name

When you want to change the softkey label names for the user menu function, you need to code within the VBA program. For details, refer to the Sample Program on Settings for Softkey Label and Softkey Enabled/Disabled (object name: Module1).

For more information on this object, see:

- COM Object Model
- SCPI.PROGram.SKEY.ITEM(1-8).LABel

Enabling / Disabling Softkey

When you want to enable or disable the softkey for the user menu function, you need to code within the VBA program. For details, refer to the Sample Program on Settings for Softkey Label and Softkey Enabled/Disabled (object name: Module1).

How to use the User Menu Function

To execute the procedure assigned to a softkey, you need to generate an event by pressing the softkey.

For this, you need to code within the VBA program to execute a procedure. For more information, refer to the Sample Program on Executing User Menu (object name: Class1).

Use the following function to execute a procedure in the user menu:

1. Enable the softkey to generate an event using **Macro Setup** > **E5052 Event** ON.
2. Select **Macro Setup** > **User Menu <No>**.

<No.> represents the button number. You can set the label for **<No.>** as you like. For details, refer to the Enabling/Disabling Softkey.

NOTE You can use the user menu function only when the VBA program is running. For information on how to verify whether the VBA is running, refer to Running a VBA Program.

Simple usage example

The following is a simple example that uses a standard module and a class module.

Object name	Module type	Function
Module1	Standard module	Sets the softkey labels and enables interrupts from the softkeys
Class1	Class module	Specifies the processing to be followed when an event occurs

The program (object name: Module1) is described in detail below:

Line 60

The instance of the class module specified by Class1 is assigned to clsEvent (Object creation).

Lines 80 to 140

Set the first and second softkey (*id*: 1 to 2) to enabled and set the third to eighth softkey (*id*: 3 to 8) to disabled.

Lines 160 to 170

Set the first softkey label (*id*: 1) to "Preset" and the second softkey label (*id*: 2) to "Exit".

Lines 190 to 210

Processing repeated until the event occurs.

Line 200

Detects an event when a specific softkey is pressed and enables the interrupt from the event.

Sample Program of Settings for Softkey Label and Softkey Enabled/Disabled (object name: Module1)

```
10| Sub Main()
20|
```

```
30| Dim clsEvent As Class1
40| Dim I As Long, J As Long
50|
60| Set clsEvent = New class1
70|
80| For I = 1 To 2
90| SCPI.PROGram.SKEY.Item(I).ENABle = True
100| Next I
110|
120| For J = 3 To 8
130| SCPI.PROGram.SKEY.Item(J).ENABle = False
140| Next J
150|
160| SCPI.PROGram.SKEY.Item(1).LABel = "Preset"
170| SCPI.PROGram.SKEY.Item(2).LABel = "Exit"
180|
190| Do
200| DoEvents
210| Loop
220|
230| Set clsEvent = Nothing
240|
250| End Sub
```

The program (object name: Class1) is described in detail below:

Lines 50 to 130

An event occurs when the softkey is pressed from the user menu.

Lines 70 to 80

E5052B is reset to the initial setting when the first softkey is pressed.

Lines 90 to 120

The program ends with a message box displayed when the second softkey is pressed.

Lines 140 to 150

An event occurs when the triggered sweep ends.

Line 150

A message box appears when the sweep ends.

Lines 190 to 210

Implement the object.

Lines 230 to 250

Release the object.

Sample Program Executing User Menu (object name: Class1)

```
10| Dim WithEvents app As Application
20|
30| Private Sub app_OnEvent(ByVal ReasonStr As String, ByVal OptionStr As String)
40| Select Case ReasonStr
50| Case "UserLabel"
60| Select Case OptionStr
70| Case 1
80| SCPI.SYSTem.PRESet
90| Case 2
100| SCPI.PROGram.COM.EVENt = False
110| MsgBox "Program ended"
120| End
130| End Select
140| Case "SweepEnd"
150| MsgBox "Sweep ended"
160| End Select
170| End Sub
180|
190| Private Sub Class_Initialize()
200| Set app = Application
210| End Sub
220|
```

```
230| Private Sub Class_Terminate()
240| Set app = Nothing
250| End Sub
```

The E5052B's VBA program is executed as a part of E5052B's applications. Therefore, if any executed VBA program takes a long time before returning control to Windows, all processing operations other than VBA are suspended during this time. To prevent this, use DoEvents to return the control to E5052B applications.

Argument for event occurrence

The arguments for event occurrence are described below. An event represents **app_OnEvent**, which is described in Simple usage example.

Event	First argument	Second argument
User menu	UserLabel	Softkey label NO. (1-8)
End of sweep	SweepEnd	Measurement window (AM/BB/FP/PN/SP/TR/PS/USER)
Request for service	RQS	Nothing

NOTE OnEvent(id1 As String, id2 As String) is a event handler of the events from the application and can refer to the instance (data) of the class. By declaring the object a variable "app" in the class module, it can be utilized as a procedure to obtain the event occurrence.

NOTE Within the event handler (lines 30-170), the processing should be the minimum necessary before an event ends. Nesting is also not allowed.

Limit Test

- Overview
- Using Commands to Define Limit Lines
- Example of Using Commands to Define Limit Lines
- Reading Limit Lines from Files
- Example of Reading Limit Lines from Files

Other topics about Controlling E5052B

Overview

This section describes how to define the limit lines and determine pass or fail with regard to the limit test function. For more on the concept of the limit test, refer to Data Analysis and Result Output.

Using Commands to Define Limit Lines

You can define the limit lines by specifying any limit value to the parameter of SCPI commands.

The program is described in detail below:

Line 20 to 30

Define the data array (variable) for the limit lines.

Line 40 to 50

Redefine the data array (variable) depending on the number of limit lines (segments).

Line 70 to 140

Set the upper limit value for the limit lines in the array.

Line 160 to 190

Set the lower limit value for the limit lines in the array.

Line 220 to 230

Specify the segment number for the upper and lower limit values.

Line 250 to 260

Specify the upper and lower limit values on the trace.

Line 280 to 320

Trigger the instrument.

Line 340

Display the limit lines.

Line 350

Display the determination result. (For fail only.)

Line 360

Activate the limit test function.

Example of Using Commands to define *Limit Lines*

```
10| Sub Main()
20| Dim Udata() As Double
30| Dim Ldata() As Double
40| ReDim Udata(7)
50| ReDim Ldata(3)
60|
70| Udata(0) = 0
80| Udata(1) = 1400000000#
90| Udata(2) = 0.00002
100| Udata(3) = 1400000000#
110| Udata(4) = 0.00002
120| Udata(5) = 1600000000#
130| Udata(6) = 0.0001
140| Udata(7) = 1600000000#
150|
160| Ldata(0) = 0
170| Ldata(1) = 1200000000#
180| Ldata(2) = 0.0001
190| Ldata(3) = 1200000000#
200|
210| SCPI.DISPlay.WINDow.ACTive = "FP1"
220| SCPI.CALCulate.FP.TRACe.LIMit.UPPer.SEGMent.Count = 2
230| SCPI.CALCulate.FP.TRACe.LIMit.LOWer.SEGMent.Count = 1
```

```
240|
250| SCPI.CALCulate.FP.TRACe.LIMit.UPPer.SEGMent.DATA = Udata
260| SCPI.CALCulate.FP.TRACe.LIMit.LOWer.SEGMent.DATA = Ldata
270|
280| SCPI.TRIGger.MODE = "FP1"
290| SCPI.TRIGger.FP.Source = "bus"
300| SCPI.INITiate.FP.CONTinuous = False
310| SCPI.INITiate.FP.IMMediate
320| SCPI.IEEE4882.TRG
330|
340| SCPI.DISPlay.FP.TRACe.LIMit.LINE = True
350| SCPI.DISPlay.FP.LIMit.FSIGn = True
360| SCPI.CALCulate.FP.TRACe.LIMit.STATe = True
370| End Sub
```

Reading Limit Lines from Files

You can create any upper and lower value for limit lines in the specified format in advance, and read the file later to specify the limit lines.

For information about creating upper and lower values for limit lines, refer to Data analysis and Result Output.

The file must be saved in the CSV format (with the extension *.csv).

The program is described in detail below:

Save the upper limit values together into one file, and the lower limit values into another.

Line 20

Read the lower limit value for the limit lines from the file.

Line 30

Read the upper limit value for the limit lines from the file.

Line 50 to 100

Trigger the instrument.

Line 120

Display the limit lines.

Line 130

Display the determination result. (For fail only.)

Line 140

Activate the limit test function.

Example of Reading Limit Lines from File

```
010| Sub Main()
020| SCPI.MMEMory.FP.TRACe.Load.LIMit.LOWer = "f:\lower.csv"
030| SCPI.MMEMory.FP.TRACe.Load.LIMit.UPPer = "f:\upper.csv"
040|
050| SCPI.DISPlay.WINDow.ACTive = "FP1"
060| SCPI.TRIGger.MODE = "FP1"
070| SCPI.TRIGger.FP.Source = "bus"
080| SCPI.INITiate.FP.CONTinuous = False
090| SCPI.INITiate.FP.IMMediate
100| SCPI.IEEE4882.TRG
110|
120| SCPI.DISPlay.FP.TRACe.LIMit.LINE = True
130| SCPI.DISPlay.FP.LIMit.FSIGn = True
140| SCPI.CALCulate.FP.TRACe.LIMit.STATe = True
150| End Sub
```

Reading / Writing Measurement Data

This section describes how to process the E5052B's internal data. You can use amongst the following internal data arrays: unformatted data arrays, unformatted memory arrays, formatted data arrays, formatted memory arrays, and X-axis data arrays. For more information on the internal data arrays, see Internal Data Processing.

To read/write a formatted data array, formatted memory array, unformatted data array, or unformatted memory array, use the following objects:

- SCPI.CALCulate.xx.TRACe.DATA.FDATa
- SCPI.CALCulate.xx.TRACe.DATA.FMEMory
- SCPI.CALCulate.xx.TRACe.DATA.UDATa

- SCPI.CALCulate.xx.TRACe.DATA.UMEMory

There are unformatted data array and memory array where the AM noise measurement, the baseband noise measurement, phase noise measurement and segment phase noise measurement are contained with the unit of dBc.

- SCPI.CALCulate.AM(1-1).TRACe(1-1).DATA.PDATa
- SCPI.CALCulate.AM(1-1).TRACe(1-1).DATA.PMEMory
- SCPI.CALCulate.BB(1-1).TRACe(1-1).DATA.PDATa
- SCPI.CALCulate.BB(1-1).TRACe(1-1).DATA.PMEMory
- SCPI.CALCulate.PN(1-1).TRACe(1-1).DATA.PDATa
- SCPI.CALCulate.PN(1-1).TRACe(1-1).DATA.PMEMory
- SCPI.CALCulate.PS(1-1).TRACe(1-1).DATA.PDATa
- SCPI.CALCulate.PS(1-1).TRACe(1-1).DATA.PMEMory

To read an X-axis data array, use the following object:

- SCPI.CALCulate.xx.DATA.XDATa

For the transient measurement, you can use two types of object as follows:

- SCPI.CALCulate.TR(1-1).NARRow(1-2).DATA.XDATa
- SCPI.CALCulate.TR(1-1).WIDE.DATA.XDATa

To read a raw data array, use the following object:

- SCPI.CALClate.xx.DATA.RDATa

For the transient measurement, you can use two types of object as follows:

- SCPI.CALCulate.TR(1-1).NARRow(1-2).DATA.RDATa
- SCPI.CALCulate.TR(1-1).WIDE.DATA.RDATa

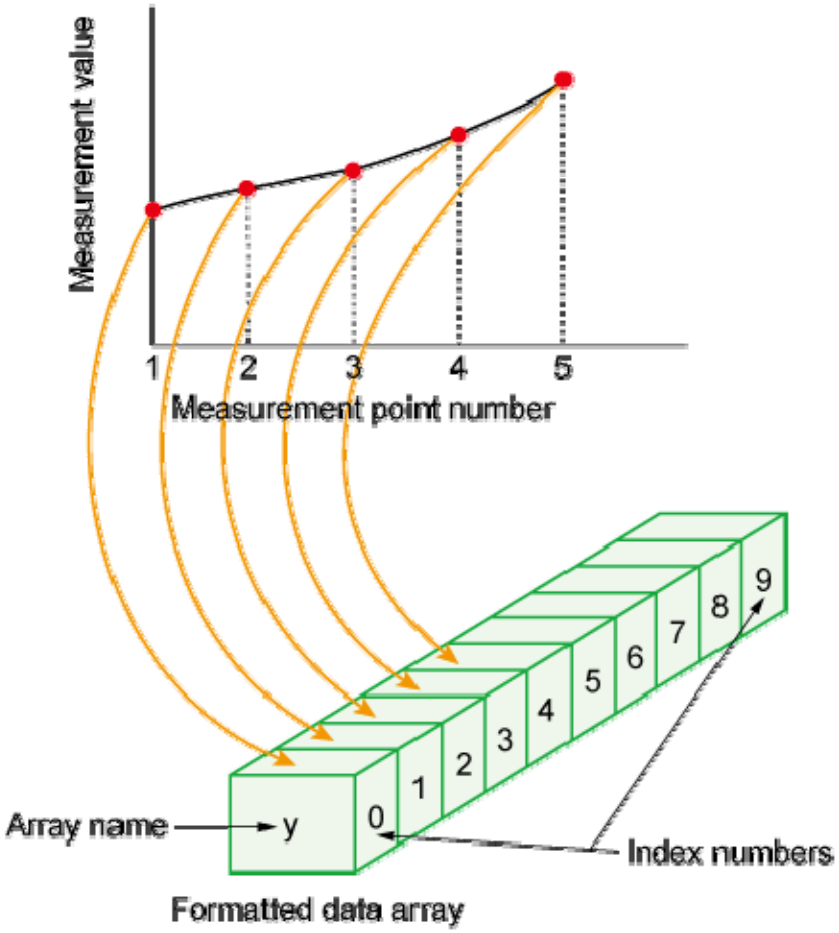
There is a raw array where the AM noise measurement, the baseband noise measurement, phase noise measurement and segment phase noise measurement are contained with the unit of dBc.

- SCPI.CALCulate.AM(1-1).DATA.PDATa
- SCPI.CALCulate.BB(1-1).DATA.PDATa
- SCPI.CALCulate.PN(1-1).DATA.PDATa

- SCPI.CALCulate.PS(1-1).DATA.PDATa

The E5052B VBA allows you to deal with multiple pieces of data through variables of the Double-precision Dynamic Array type. For example, a formatted data array that includes five measurement points is stored as shown in the figure below. For more information on contained data, see the section on “Reading/Writing Measurement Data”.

Example of storing data into a Variant variable



ssa0075

NOTE

When you use one of the objects listed above, the base index number of the array is always 0 even if the declaration section contains the "Option Base 1" statement, which specifies the use of the base array index of 1.

Controlling Peripherals

- Overview
- Programming with VISA

Overview

- Overview
- Preparation

Other topics about Controlling Peripherals

Overview

The E5052B macro function (E5052B VBA) can be used not only to automate measurements but also to control external measurement instruments connected via USB/GPIB interface by acting as a self-contained system controller (see An Overview of a Control System Based on the Macro Function).

The E5052B macro function (E5052B VBA) performs communications via the COM interface when controlling the E5052B itself, but it communicates via VISA (Virtual Instrument Software Architecture) when controlling external measurement instruments.

Preparation

Importing Definition Files

To use the VISA library in the E5052B macro (E5052B VBA), you need to import two definition files into your project with the Visual Basic editor to define the VISA functions and perform other tasks. The definition files are stored on the sample programs disk under the following filenames (for information on importing modules, refer to Saving a Module (Exporting).

- **visa32.bas**
- **vpptype.bas**

Programming with VISA

- Overview
- Starting VISA
- Connection
- Communication
- Disconnection
- Sample Program

Other topics about Controlling Peripherals

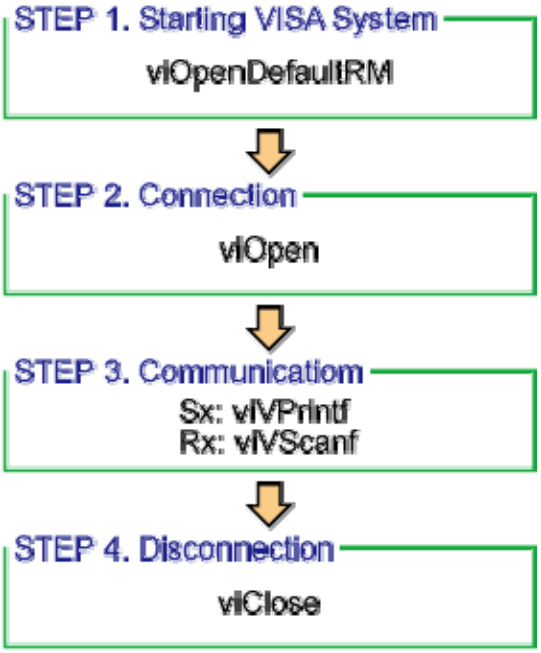
Overview

The following figure shows the flow of controlling the instrument with VISA. When developing a VISA program in the Visual Basic language, a special programing notice (in the readme text filie listed below) must be reviewed.

For details on the use of the VISA library and the programing notice for using the VISA library with the E5052B macro (E5052B VBA), refer to the following files contained in IO library CD-ROM.

- **visa.hlp** (on-line help for the VISA library)
- **vbreadme.txt** (notes on using the VISA library with VB)

Flow of instrument control with VISA



ssa0077

STEP 1. Starting VISA

The VISA system startup session is processed in Line 90 in the sample program ctrl_ext.vba. VISA's viOpenDefaultRM function initializes and starts up the VISA system. The viOpenDefaultRM function must be executed before other VISA functions are called, and the parameter of this function is startup information (Defrm in ctrl_ext.vba).

Syntax

viOpenDefaultRM(*param*)

Parameter

Parameter	(*param*)
Description	Startup information (output)
Data type	Long integer type

STEP 2. Connection

The connection session is handled in Line 130. VISA's viOpen function makes connection with the specified instrument. The viOpen function returns a value so that the VISA functions can apply it to the specified instrument. The parameters of this function are startup information (Defrm), the address information of the specified instrument ("GPIB0::17::INSTR" in **ctrl_ext.vba**), access mode (0 in **ctrl_ext.vba**), timeout (0 in **ctrl_ext.vba**), and connection information (Equip in **ctrl_ext.vba**).

Syntax

viOpen(*param1*, *param2*, *param3*, *param4*, *param5*)

Parameters

Parameter	(*param1*)
Description	Startup information (input)
Data type	Long integer type

Parameter	(*param2*)
Description	Address information of the specified instrument (input)
Data type	Character string type
Syntax	"GPIB0::*gpib address*::INSTR" "USB0::*manufacturer ID*::*model code*::*serial number*::0::INSTR" (ex. "USB0::2391::2312::MY12345678::0::INSTR") "TCPIP0:: *IP address*::inst0::INSTR"

Parameter	(*param3*)
Description	Access mode (*Enter* 0)

Parameter	(*param4*)
Description	Timeout (Enter 0)

Parameter	(*param5*)
Description	Connection information (output)
Data type	Long integer type

STEP 3. Communication

The communication session is conducted in Line 170. VISA's viVPrintf function sends a program message (*GPIB* command) to the specified instrument. The parameters of this function are connection information (Equip), the program message (*IDN?), and the variable to be formatted (0 in **ctrl_ext.vba**).

NOTE

To input/output GPIB commands, the viVPrintf function and the viVScanf function are mainly used, but other VISA functions are also available. For more information, refer to **visa.hlp** (online help for the VISA library).

Syntax

viVPrintf(*param1*, *param2*, *param3*)

Parameters

Parameter	(*param1*)
Description	Connection information (input)
Data type	Long integer type

Parameter	(*param2*)
Description	Program message (input) When sending a program message of the GPIB command, a message terminator is required at the end of the message (Chr$(10) in **ctrl_ext.vba**)

Data type	Character string type

Parameter	(*param3*)
Description	A variable to be formatted. If not applicable, enter 0.
Data type	Specified data type

The receiving session is controlled in Line 210. VISA's viVScanf function receives the result from the specified instrument and stores it in the output variable. The parameters of this function are connection information (Equip in **ctrl_ext.vba**), the format parameter for the output variable (%t in **ctrl_ext.vba**), and the output variable (Prod in **ctrl_ext.vba**).

Syntax

viVScanf(*param1*, *param2*, *param3*)

Parameters

Parameter	(*param1*)
Description	Connection information (input)
Data type	Long integer type

Parameter	(*param2*)
Description	Format parameter for the output variable
Data type	Character string type

Parameter	(*param3*)
Description	Output variable (output)
Data type	Character string type

STEP 4. Disconnection

The disconnection session is handled in Line 280. VISA's viClose function disconnects communication and terminates the VISA system. The parameter of this function is startup information (Defrm in **ctrl_ext.vba**).

Syntax

viClose(*param*)

Parameter

Parameter	(*param*)
Description	Startup information (input)
Data type	Long integer type

Sample Program to Read Out the Product Information of Peripheral (Instrument)

The **ctrl_ext.vba** is a sample program to control instruments connected through USB/GPIB interface cable using the E5052B as the system controller. This VBA program consists of the following modules.

Object name	**Module type**	**Content**
mdlVisa	Standard module	Reads out the product information of external instrument.
Module1 Module2	Standard module	Two definition files to use VISA library

NOTE

When you control peripherals from E5052B VBA, use the GPIB commands provided for the instrument to communicate over VISA. On the other hand, when you control the E5052B itself from E5052B VBA, use the COM objects provided for the E5052B to communicate.

Lines 90 to 100

Initializes and starts up the VISA system and outputs the startup information to the Defrm variable. During this process, if an error occurs, the program goes to the error handling routine (Lines 320 to 360).

Lines 130 to 140

Establishes the connection to the external instrument (GPIB address: 17) connected via GPIB and outputs the connection information to the Equip variable. During this process, if an error occurs, the program goes to the error handling routine (Lines 320 to 360).

Lines 170 to 180

Queries the product information of the external instrument connected via USB/GPIB interface cable using VISA. During this process, if an error occurs, the program goes to the error handling routine (Lines 320 to 360).

Lines 210 to 250

Retrieves the product information through VISA and outputs it into the Prod variable. Displays the read-out result in the message box. During this process, if an error occurs, the program goes to the error handling routine (Lines 320 to 360).

Line 280

Breaks the communication and terminates the VISA system.

Lines 320 to 360

If an error occurs in a VISA function, displays the detail of the error and terminates the program.

Read out the product information (ctrl_ext.vba)

```
10| Sub Main()
20|
30| Dim status As Long 'VISA function status return code
40| Dim Defrm As Long 'Session to Default Resource Manager
50| Dim Equip As Long 'Session to instrument
60| Dim Prod As String * 100 'String to receive the result
70|
80| ' Initializes the VISA system.
90| status = viOpenDefaultRM(Defrm)
100| If (status <> VI_SUCCESS) Then GoTo VisaErrorHandler
```

```
110|
120| ' Opens the session to the specified instrument.
130| status = viOpen(Defrm, "GPIB0::17::INSTR", 0, 0, Equip)
140| If (status <> VI_SUCCESS) Then GoTo VisaErrorHandler
150|
160| ' Asks for the instrument's product information.
170| status = viVPrintf(Equip, "*IDN?" & Chr$(10), 0)
180| If (status <> VI_SUCCESS) Then GoTo VisaErrorHandler
190|
200| ' Reads the result.
210| status = viVScanf(Equip, "%t", Prod)
220| If (status <> VI_SUCCESS) Then GoTo VisaErrorHandler
230|
240| ' Displays the result.
250| MsgBox Prod
260|
270| ' Closes the resource manager session (which closes everything)
280| Call viClose(Defrm)
290|
300| GoTo Prog_end
310|
320| VisaErrorHandler:
330| Dim VisaErr As String * 200
340| Call viStatusDesc(Defrm, status, VisaErr)
350| MsgBox "Error : " & VisaErr, vbExclamation
360| Exit Sub
370|
380| Prog_end:
390|
400| End Sub
```

User Defined Window

Overview

The E5052B's user defined window provides graphics utility for tailoring the measurement and interpreting the results. Users can operates scale, trace, and marker functions as same as that of E5052B's other instrument mode. This section contains:

- Analysis Functions and Save/Recall Functions
- How to use the User Defined Window

Analysis Functions and Save / Recall Functions

- Analyzing Data on the Trace Using the Marker
- Searching for Positions that Match Specified Criteria
- Determining the Mean, Standard Deviation, and p-p of the Trace
- Comparing Traces/ Performing Data Math
- Saving and Recalling Instrument State
- Saving Trace Data to a File

Other topics about User Defined Window

How to use the User Defined Window

- Overview
- Printing Measurement Data in the User Define Window
- Example of Measurement Data in the User Defined Window

Other topics about User Defined Window

Overview

This section explains how to use the user defined window on the E5052B.

Printing Measurement Data in the User Defined Window

The E5052B's user defined window enables the display traces by copying the data array to the trace array of the user defined window. Users can access all the data array of the user defined window via either VBA COM commands or SCPI commands. Up to 8 traces can be displayed in the E5052B's user defined window.

The following example shows a sample procedure that demonstrates how to display traces that users define data array both in X-axis and Y-axis.

Lines 20 to 30

Defines data array to be displayed in the trace of user defined window.

Line 50

Selects and specifies the user defined window as active window.

Lines 80 to 90

Copies formatted data trace from the frequency-power measurement results to the data array defined in the VBA program.

Line 110

Specifies trace 1 as active trace

Line 140

Copies data array to X-axis data on trace 1

Line 170

Copies data array to Y-axis data on trace 1

Lines 200 to 210

Sets display unit of X-axis and Y-axis respectively.

Line 240

Execute autoscale

Line 260

Returns to the E5052B application

Example of Measurement Data in the User Defined Window

```
10| Sub Main()
20| Dim aryXdata() As Double
30| Dim aryYdata() As Double
40|
50| SCPI.DISPlay.USER.STATe = True
```

```
60| SCPI.DISPlay.WINDow.ACTive = "USER1"
70|
80| aryXdata = SCPI.CALCulate.FP.DATA.XDATa
90| aryYdata = SCPI.CALCulate.FP.TRACe(1).DATA.FDATa
100|
110| SCPI.DISPlay.USER.TRACe(1).STATe = True
120|
130| 'x data
140| SCPI.CALCulate.USER.TRACe(1).DATA.XDATa = aryXdata
150|
160| 'y data
170| SCPI.CALCulate.USER.TRACe(1).DATA.FDATa = aryYdata
180|
190| 'Unit
200| SCPI.DISPlay.USER.TRACe(1).X.UNIT = "V"
210| SCPI.DISPlay.USER.TRACe(1).Y.UNIT = "Hz"
220|
230| 'Auto scale
240| SCPI.DISPlay.USER.ALLTrace.Y.SCALe.AUTO
250|
260| DoEvents
270|
280| End Sub
```

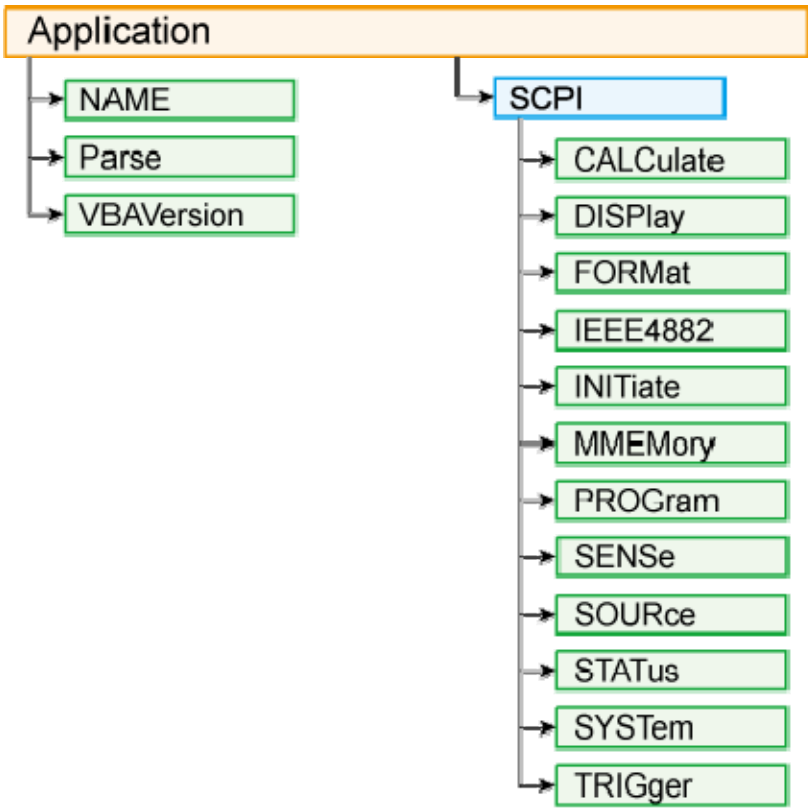
The E5052B's VBA program is executed as an application. Therefore, if any executed VBA program takes a long time before returning control to the E5052B, all processing operations other than VBA are suspended during this time. To prevent this, use DoEvents to return the control to the E5052B.

The value in the X-data array for the user defined window has to have incremental order, that means (the value at N point) < (the value at N+1 point). When the error message, "Illegal parameter value" is received, check the values of the X-data array data on point-to-point to ensure this rule.

Command Reference

COM Object Model

The COM objects provided for the E5052B are structured hierarchically as shown below.



ssa0078

Application Objects

The Application objects are at the top of the hierarchy of the E5052B COM object model. They consist of 3 objects dedicated to the COM interface and SCPI objects corresponding to SCPI commands.

SCPI Objects

The SCPI objects are created to realize the SCPI commands of the E5052B with the COM interface.

The conversion rules from the SCPI commands when writing SCPI object messages are as follows:

- SCPI. must be at the beginning. Notice that the IEEE common commands start with SCPI.IEEE4882. and "*" is omitted.
- Replace colons (:) used as the hierarchical separator symbol with dots (.).
- The number written in the object message is specified with ( ).
- You cannot omit the command message in the syntax.

SCPI command	COM object
OUTPUT 717;":SENS:SP:AVER:CONT 16"	SCPI.SENSe.SP.AVERage.COUNt = 16
OUTPUT 717;":SENS:SP:AVER:STAT?" ENTER 717;A$	A = SCPI.SENSe.SP.AVERage.STATe
OUTPUT 717;"*CLS"	SCPI.IEEE4882.CLS

Notational Conventions

This section describes the notational conventions used for the description of commands reference.

Object Type

Object type describes different types of E5052B COM objects. The E5052B provides properties and methods as COM objects. COM objects which set (send)/read (return) the state of the E5052B using variables are defined as property and COM objects which does other processing are defined as method.

COM objects used only to read the state of the E5052B are indicated with "**Read-only**" and ones used only to set the state of the E5052B are indicated by "**Write-only**" . COM object that can both read and write data to the E5052B are indicated by '**Read-Write**'.

Syntax

Syntax describes the syntax for sending a COM object from the E5052B VBA to the E5052B. The syntax consists of two parts: the object part and the set/read part, with an equal "=" inserted between them. Variables are indicated by italicized letters. Variables with () are indices. For indices with () having their preset values, you can omit "(*variable*)," and, if omitted, the preset values are automatically set.

The following table describes the 3 types of syntax for coding using objects:

Type	Description
"Object (property) = *variable*":	Set the stat of the E5052B.
variable=object (property):	Read the stat of the E5052B.
"Object (method)":	Perform some processing in the E5052B.

Description

Description describes how to use the COM object or the operation when executed.

Type	Description
Ch	Number of channels. For all measurement menus the number of channel is 1.
Tr	Number of traces. NOTE For **PN**, **SP**, **AM**, **PS** and **BB**, number of trace is 1. For **FP**, number of trace is 3. For **TR**, number of trace is 8.
Mk	Number of markers. For all measurement menus the number of marker is 10.
Nr	Number of narrow is 2. NOTE **TR** use only.

Variable

Variable provides description about different variables that can be used with the COM objects. It gives the description, data type, allowable range, preset value, unit, resolution, and notes for *variable (italic)* shown in the syntax.

NOTE Variables declared as the string data type (String) are not case-sensitive. For variables of the string type that indicate arguments (written as *Param* in the syntax), you can omit lower-case letters.

The data types of the E5052B COM objects include 5 types as shown in the following table. Before using variables, declare the data type of each variable. If you do not declare the data type of a variable, it is automatically processed as a variant type.

Data type	Name	Consumed memory	Range
Long	Long integer type	4 bytes	-2,147,483,648 to 2,147,483,647
Double	Double precision floating point type	8 bytes	For a negative value: -1.79769313486232E+308 to -4.94065645841247E-324 For a positive value: -1.79769313486232E+308 to -4.94065645841247E-324
Boolean	Boolean type	2 bytes	For COM: True or False (For SCPI: ON or OFF)
String	Character string type	1 byte / alphanumeric character	Up to approximately 2 billion characters
Variant	Variant type	16 bytes	No limitation

Examples

Examples provides a sample example of using the object through coding with the E5052B VBA.

Related Objects

Related objects provides information about other objects that are similar/related with the object.

Equivalent Key

Equivalent key shows the operational procedure of the front panel keys that has the same effect as this object.

Equivalent SCPI command

Equivalent SCPI command shows the SCPI command to execute from an external controller. Its syntax, query response, and example of use are provided.

IEEE

SCPI.IEEE4882.CLS

Object Type

Method (**Write Only**)

Syntax

SCPI.IEEE4882.CLS

Description

This command clears Status.

Examples

SCPI.IEEE4882.CLS

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

*CLS

Example of use

10 OUTPUT 717;"*CLS"

SCPI.IEEE4882.ESE

Object Type

Property (**Read-Write**)

Syntax

SCPI.IEEE4882.ESE = *Value*

Value = SCPI.IEEE4882.ESE

Description

This command sets/gets the value of Standard Event Status Enable Register.

Variable

Parameter	*Value*
Description	Value of the Standard Event Status Enable Register
Data Type	Long integer type (Long)
Range	0 ~ 255
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.IEEE4882.ESE = Var
Var = SCPI.IEEE4882.ESE
```

Related Objects

SCPI.IEEE4882.SRE

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

*ESE {0 ~ 255}

*ESE?

Query Response

{0 ~ 255} <newline><^END>

Example of use

```
10 OUTPUT 717;"*ESE 0 "
20 OUTPUT 717;"*ESE ?"
30 ENTER 717;A
```

SCPI.IEEE4882.ESR

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.IEEE4882.ESR

Description

This command gets the value of Standard Event Status Register.

Variable

Parameter	*Value*
Description	Value of the Standard Event Status Register
Data Type	Long integer type (Long)
Range	-
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var = SCPI.IEEE4882.ESR
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

*ESR?

Query Response

{Long} <newline><^END>

Example of use

```
10 OUTPUT 717;"*ESR ?"
20 ENTER 717;A
```

SCPI.IEEE4882.IDN

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.IEEE4882.IDN

Description

This command gets the product information (manufacturer, model number, serial number, and firmware version number) of the E5052B.

Variable

Parameter	*Value*
Description	Product information
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as String
Var = SCPI.IEEE4882.IDN
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

*IDN?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;"*IDN ?"
20 ENTER 717;A$
```

SCPI.IEEE4882.LRN

Object Type

Property (**Read Only**)

Syntax

SCPI.IEEE4882.LRN

Description

This command gets device setup query.

Examples

SCPI.IEEE4882.LRN

Related Objects

SCPI.IEEE4882.IDN

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

*LRN?

Example of use

```
10 OUTPUT 717;"*LRN ?"
```

SCPI.IEEE4882.OPC

Object Type

Property (**Read-Write**)

Syntax

SCPI.IEEE4882.OPC = *Value*

Value = SCPI.IEEE4882.OPC

Description

This command sets/gets Operation complete query.

NOTE OPC command in SCPI has no parameter. SCPI.IEEE4882.OPC command has a dummy parameter.".

Variable

Parameter	*Value*
Description	Operation complete query
Data Type	Long integer type (Long)
Range	-
Preset Value	1
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
SCPI.IEEE4882.OPC = Var
Var = SCPI.IEEE4882.OPC
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

*OPC

*OPC?

Query Response

{Long} <newline><^END>

Example of use

```
10 OUTPUT 717;"*OPC -"
20 OUTPUT 717;"*OPC ?"
30 ENTER 717;A
```

SCPI.IEEE4882.OPT

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.IEEE4882.OPT

Description

This command gets model option.

Variable

Parameter	*Value*
Description	Model option
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as String
Var = SCPI.IEEE4882.OPT
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

*OPT?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;"*OPT ?"
20 ENTER 717;A$
```

SCPI.IEEE4882.RST

Object Type

Method (**Write Only**)

Syntax

SCPI.IEEE4882.RST

Description

This command resets instrument state.

Examples

SCPI.IEEE4882.RST

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

*RST

Example of use

10 OUTPUT 717;"*RST"

SCPI.IEEE4882.SRE

Object Type

Property (**Read-Write**)

Syntax

SCPI.IEEE4882.SRE = *Value*

Value = SCPI.IEEE4882.SRE

Description

This command sets/gets service request enable register.

NOTE Only bit[0-5,7] are used. When query, bit 6 is always 0.

Variable

Parameter	*Value*
Description	Service request enable register
Data Type	Long integer type (Long)
Range	0 ~ 255
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.IEEE4882.SRE = Var
Var = SCPI.IEEE4882.SRE
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

*SRE {0 ~ 255}

*SRE?

Query Response

{0 ~ 255} <newline><^END>

Example of use

```
10 OUTPUT 717;"*SRE 0 "
20 OUTPUT 717;"*SRE ?"
30 ENTER 717;A
```

SCPI.IEEE4882.STB

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.IEEE4882.STB

Description

This command gets status byte register. 8 bits..

Variable

Parameter	*Value*
Description	Status byte register
Data Type	Long integer type (Long)
Range	-
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var = SCPI.IEEE4882.STB
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

*STB?

Query Response

{Long} <newline><^END>

Example of use

```
10 OUTPUT 717;"*STB ?"
20 ENTER 717;A
```

SCPI.IEEE4882.TRG

Object Type

Method (**Write Only**)

Syntax

SCPI.IEEE4882.TRG

Description

This command sets BUS Trigger.

Examples

SCPI.IEEE4882.TRG

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

*TRG

Example of use

```
10 OUTPUT 717;"*TRG"
```

SCPI.IEEE4882.TST

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.IEEE4882.TST

Description

This command gets Self-Test Query. During the TST? command execution, the cursor of mouse shows an hourglass. From Firmware ver **A.03.10**, this command executes in the following manner:

- Resets (*RST) the E5052B. However, the display is not renewed.
- Executes the E5052B Self test (same as the Power On test).
- Executes the E5053A Self test (if E5053A is connected i.e. the power is On, and USB is connected)
- Resets (*RST) the E5052B again.

Depending on the Self test result, this command return the values:

- 0: Pass (Both E5052B and E5053A pass)
- 1: Fail (If either one of them fails, to be Fail)

NOTE When E5053A is not connected, this command determine the result of E5052B alone.

NOTE Do not turn off the E5053A power supply and disconnect the USB during the *TST? command execution when E5053A is connected.

NOTE If E5053A is disconnected during the measurement, the test time becomes longer and the test result may Fail. The E5053A should be disconnected before or after the *TST command.

Variable

Parameter	*Value*

Description	Self-Test Query
Data Type	Long integer type (Long)
Range	0 or 1
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var = SCPI.IEEE4882.TST
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

*TST?

Query Response

{Long} <newline><^END>

Example of use

```
10 OUTPUT 717;"*TST ?"
20 ENTER 717;A
```

SCPI.IEEE4882.WAI

Object Type

Method (**Write Only**)

Syntax

SCPI.IEEE4882.WAI

Description

This command sets Wait to Continue Command.

Examples

SCPI.IEEE4882.WAI

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

*WAI

Example of use

10 OUTPUT 717;"*WAI"

Abort

SCPI.ABORt

Object Type

Method (**Write Only**)

Syntax

SCPI.ABORt

Description

This command aborts current sweep.

Examples

SCPI.ABORt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:ABORt

Example of use

```
10 OUTPUT 717;":ABORt"
```

Calculate

SCPI.CALCulate.AM(Ch).ALLTrace.LIMit.FAIL

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.AM(Ch).ALLTrace.LIMit.FAIL

Description

This command gets limit test pass/fail, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Result of limit test
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Boolean
Var= 1
Var = SCPI.CALCulate.AM1.ALLTrace.LIMit.FAIL
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:ALLTrace:LIMit:FAIL?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:ALLTrace:LIMit:FAIL ?"
20 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.COUPle.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.COUPle.STATe = *Value*

Value = SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.COUPle.STATe

Description

This command sets/gets marker couple on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker Coupling Function state (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.AM1.ALLTrace.MARKer.COUPle.STATe = Var
Var = SCPI.CALCulate.AM1.ALLTrace.MARKer.COUPle.STATe
```

Related Objects

SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.DISCrete.STATe

SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.REFerence.NUMBer

SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.REFerence.STATe

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:ALLTrace:MARKer:COUPle:STATe {ON|OFF|1|0}

:CALCulate:AM[1-1]:ALLTrace:MARKer:COUPle:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:ALLTrace:MARKer:COUPle:STATe 1"
20 OUTPUT 717;":CALCulate:AM1:ALLTrace:MARKer:COUPle:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.DISCrete.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.DISCrete.STATe = *Value*

Value = SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.DISCrete.STATe

Description

This command sets/gets marker discrete on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Enables/Disables marker discrete
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.AM1.ALLTrace.MARKer.DISCrete.STATe = Var
Var = SCPI.CALCulate.AM1.ALLTrace.MARKer.DISCrete.STATe
```

Related Objects

SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.COUPle.STATe

SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.REFerence.NUMBer

SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.REFerence.STATe

Equivalent Key

AM Menu: **Marker** > **More Functions** > **Discrete**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:ALLTrace:MARKer:DISCrete:STATe {ON|OFF|1|0}

:CALCulate:AM[1-1]:ALLTrace:MARKer:DISCrete:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:ALLTrace:MARKer:DISCrete:STATe 1"
20 OUTPUT 717;":CALCulate:AM1:ALLTrace:MARKer:DISCrete:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.REFerence.NUMBer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.REFerence.NUMBer = *Value*

Value = SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.REFerence.NUMBer

Description

This command sets/gets reference marker number., for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker reference number
Data Type	Long integer type (Long)
Range	1 ~ 10
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.AM1.ALLTrace.MARKer.REFerence.NUMBer = Var
Var = SCPI.CALCulate.AM1.ALLTrace.MARKer.REFerence.NUMBer
```

Related Objects

SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.COUPle.STATe

SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.DISCrete.STATe

SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.REFerence.STATe

Equivalent Key

AM Menu: **Marker** > **More Functions** > **Ref Marker**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:ALLTrace:MARKer:REFerence:NUMBer {1 ~ 10}

:CALCulate:AM[1-1]:ALLTrace:MARKer:REFerence:NUMBer?

Query Response

{1 ~ 10} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:ALLTrace:MARKer:REFerence:NUMBer 1 "
20 OUTPUT 717;":CALCulate:AM1:ALLTrace:MARKer:REFerence:NUMBer ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.REFerence.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.REFerence.STATe = *Value*

Value = SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.REFerence.STATe

Description

This command sets/gets delta marker mode on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Delta marker mode (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.AM1.ALLTrace.MARKer.REFerence.STATe = Var
Var = SCPI.CALCulate.AM1.ALLTrace.MARKer.REFerence.STATe
```

Related Objects

SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.COUPle.STATe

SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.DISCrete.STATe

SCPI.CALCulate.AM(Ch).ALLTrace.MARKer.REFerence.NUMBer

Equivalent Key

AM Menu: **Marker** > **More Functions** > **Ref Marker Mode**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:ALLTrace:MARKer:REFerence:STATe {ON|OFF|1|0}

:CALCulate:AM[1-1]:ALLTrace:MARKer:REFerence:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:ALLTrace:MARKer:REFerence:STATe 1"
20 OUTPUT 717;":CALCulate:AM1:ALLTrace:MARKer:REFerence:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).DATA.CARRier

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).DATA.CARRier = *Value*

Value = SCPI.CALCulate.AM(Ch).DATA.CARRier

Description

This command sets/gets Carrier data: freq, power, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Carrier data: Freq & Power
Data Type	Variant type Array (Range)
Range	1...2
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 2) as Variant
SCPI.CALCulate.AM1.DATA.CARRier = Var
Var = SCPI.CALCulate.AM1.DATA.CARRier
```

Related Objects

SCPI.CALCulate.AM(Ch).DATA.PDATa

SCPI.CALCulate.AM(Ch).DATA.RDATa

SCPI.CALCulate.AM(Ch).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:DATA:CARRier {1...2}

:CALCulate:AM[1-1]:DATA:CARRier?

Query Response

{1...2} <newline><^END>

Example of use

```
10 Dim A(1:2)
20 OUTPUT 717;":CALCulate:AM1:DATA:CARRier 1,2"
30 OUTPUT 717;":CALCulate:AM1:DATA:CARRier ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.AM(Ch).DATA.PDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).DATA.PDATa = *Value*

Value = SCPI.CALCulate.AM(Ch).DATA.PDATa

Description

This command sets/gets measurement raw power data, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Measurement raw power data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.AM1.DATA.PDATa = Var
Var = SCPI.CALCulate.AM1.DATA.PDATa
```

Related Objects

SCPI.CALCulate.AM(Ch).DATA.CARRier

SCPI.CALCulate.AM(Ch).DATA.RDATa

SCPI.CALCulate.AM(Ch).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:DATA:PDATa {1...1601}

:CALCulate:AM[1-1]:DATA:PDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:AM1:DATA:PDATa 1,1601"
30 OUTPUT 717;":CALCulate:AM1:DATA:PDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.AM(Ch).DATA.RDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).DATA.RDATa = *Value*

Value = SCPI.CALCulate.AM(Ch).DATA.RDATa

Description

This command sets/gets measurement raw data, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Measurement raw data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.AM1.DATA.RDATa = Var
Var = SCPI.CALCulate.AM1.DATA.RDATa
```

Related Objects

SCPI.CALCulate.AM(Ch).DATA.CARRier

SCPI.CALCulate.AM(Ch).DATA.PDATa

SCPI.CALCulate.AM(Ch).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:DATA:RDATa {1...1601}

:CALCulate:AM[1-1]:DATA:RDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:AM1:DATA:RDATa 1,1601"
30 OUTPUT 717;":CALCulate:AM1:DATA:RDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.AM(Ch).DATA.XDATa

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.AM(Ch).DATA.XDATa

Description

This command gets X axis data, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	X axis data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
Var = SCPI.CALCulate.AM1.DATA.XDATa
```

Related Objects

SCPI.CALCulate.AM(Ch).DATA.CARRier

SCPI.CALCulate.AM(Ch).DATA.PDATa

SCPI.CALCulate.AM(Ch).DATA.RDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:DATA:XDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:AM1:DATA:XDATa ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).ALLMarker.ACTive

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).ALLMarker.ACTive = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).ALLMarker.ACTive

Description

This command sets/gets active marker, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Active marker
Data Type	Long integer type (Long)
Range	1 ~ 10
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.AM1.TRACe1.ALLMarker.ACTive = Var
Var = SCPI.CALCulate.AM1.TRACe1.ALLMarker.ACTive
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:ALLMarker:ACTive {1 ~ 10}

:CALCulate:AM[1-1]:TRACe[1-1]:ALLMarker:ACTive?

Query Response

{1 ~ 10} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:ALLMarker:ACTive 1 "
20 OUTPUT 717;":CALCulate:AM1:TRACe1:ALLMarker:ACTive ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

Description

This command sets/gets marker search X range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search X range source
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	FRANge
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.AM1.TRACe1.ALLMarker.SEARch.DOMain.X = Var
Var = SCPI.CALCulate.AM1.TRACe1.ALLMarker.SEARch.DOMain.X
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

SCPI.CALCulate.AM(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Equivalent Key

AM Menu: **Marker Search** > **Search Range (X)**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:ALLMarker:SEARch:DOMain:X {FRANge|BDMarker}

:CALCulate:AM[1-1]:TRACe[1-1]:ALLMarker:SEARch:DOMain:X?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:ALLMarker:SEARch:DOMain:X FRANge"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:ALLMarker:SEARch:DOMain:X ?"
30 ENTER 717;A$
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Description

This command sets/gets marker search Y range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search Y range
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	FRANge
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.AM1.TRACe1.ALLMarker.SEARch.DOMain.Y = Var
Var = SCPI.CALCulate.AM1.TRACe1.ALLMarker.SEARch.DOMain.Y
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

SCPI.CALCulate.AM(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Equivalent Key

AM Menu: **Marker Search** > **Search Range (Y)**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:ALLMarker:SEARch:DOMain:Y {FRANge|BDMarker}

:CALCulate:AM[1-1]:TRACe[1-1]:ALLMarker:SEARch:DOMain:Y?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:ALLMarker:SEARch:DOMain:Y FRANge"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:ALLMarker:SEARch:DOMain:Y ?"
30 ENTER 717;A$
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Description

This command sets search peak all, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.AM1.TRACe1.ALLMarker.SEARch.PEAK

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

SCPI.CALCulate.AM(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Equivalent Key

AM Menu: **Marker Search** > **Peak** > **Search Peak All**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:ALLMarker:SEARch:PEAK

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:ALLMarker:SEARch:PEAK"
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).ALLMarker.SEARch.SPURious

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).ALLMarker.SEARch.SPURious

Description

This command sets search spurious all, for the selected trace *Tr* of the selected channel *Ch*. You can display up to 10 markers.

NOTE This function is allowed to use when the spurious display is power.

Examples

SCPI.CALCulate.AM1.TRACe1.ALLMarker.SEARch.SPURious

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

Equivalent Key

AM Menu: **Marker Search** > **Spurious** > **Search Spurious All**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:ALLMarker:SEARch:SPURious

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:ALLMarker:SEARch:SPURious"
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.CENTer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.CENTer = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.CENTer

Description

This command sets/gets band marker X center, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X center
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.AM1.TRACe1.BDMarker.X.CENTer = Var
Var = SCPI.CALCulate.AM1.TRACe1.BDMarker.X.CENTer
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

AM Menu: **Marker Search** > **Band Marker X** > **Center**

AM Menu: **Marker Function** > **Band Marker X** > **Center**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:BDMarker:X:CENTer {-1T ~ 1T}

:CALCulate:AM[1-1]:TRACe[1-1]:BDMarker:X:CENTer?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:BDMarker:X:CENTer -10000000000"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:BDMarker:X:CENTer ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.SPAN

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.SPAN = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.SPAN

Description

This command sets/gets band marker X span, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X span
Data Type	Double precision floating point type (Double)
Range	0 ~ 2T
Preset Value	2T
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.AM1.TRACe1.BDMarker.X.SPAN = Var
Var = SCPI.CALCulate.AM1.TRACe1.BDMarker.X.SPAN
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

AM Menu: **Marker Search** > **Band Marker X** > **Span**

AM Menu: **Marker Function** > **Band Marker X** > **Span**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:BDMarker:X:SPAN {0 ~ 2T}

:CALCulate:AM[1-1]:TRACe[1-1]:BDMarker:X:SPAN?

Query Response

{0 ~ 2T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:BDMarker:X:SPAN 0"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:BDMarker:X:SPAN ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.STARt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.STARt = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.STARt

Description

This command sets/gets band marker X start, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X start
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	-1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.AM1.TRACe1.BDMarker.X.STARt = Var
Var = SCPI.CALCulate.AM1.TRACe1.BDMarker.X.STARt
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

AM Menu: **Marker Search** > **Band Marker X** > **Start**

AM Menu: **Marker Function** > **Band Marker X** > **Start**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:BDMarker:X:STARt {-1T ~ 1T}

:CALCulate:AM[1-1]:TRACe[1-1]:BDMarker:X:STARt?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:BDMarker:X:STARt -10000000000"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:BDMarker:X:STARt ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.STATe = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.STATe

Description

This command sets/gets band marker visible on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker visibility (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.AM1.TRACe1.BDMarker.X.STATe = Var
Var = SCPI.CALCulate.AM1.TRACe1.BDMarker.X.STATe
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

AM Menu: **Marker Search** > **Band Marker X** > **Band Marker X**

AM Menu: **Marker Function** > **Band Marker X** > **Band Marker X**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:BDMarker:X:STATe {ON|OFF|1|0}

:CALCulate:AM[1-1]:TRACe[1-1]:BDMarker:X:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:BDMarker:X:STATe 1"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:BDMarker:X:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.STOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.STOP = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.STOP

Description

This command sets/gets band marker X stop, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X stop
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.AM1.TRACe1.BDMarker.X.STOP = Var
Var = SCPI.CALCulate.AM1.TRACe1.BDMarker.X.STOP
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.X.STATe

Equivalent Key

AM Menu: **Marker Search** > **Band Marker X** > **Stop**

AM Menu: **Marker Function** > **Band Marker X** > **Stop**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:BDMarker:X:STOP {-1T ~ 1T}

:CALCulate:AM[1-1]:TRACe[1-1]:BDMarker:X:STOP?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:BDMarker:X:STOP -10000000000"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:BDMarker:X:STOP ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.CENTer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.CENTer = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.CENTer

Description

This command sets/gets band marker Y center, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y center
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.AM1.TRACe1.BDMarker.Y.CENTer = Var
Var = SCPI.CALCulate.AM1.TRACe1.BDMarker.Y.CENTer
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

AM Menu: **Marker Search** > **Band Marker Y** > **Center**

AM Menu: **Marker Function** > **Band Marker Y** > **Center**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:BDMarker:Y:CENTer {-1T ~ 1T}

:CALCulate:AM[1-1]:TRACe[1-1]:BDMarker:Y:CENTer?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:BDMarker:Y:CENTer -10000000000"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:BDMarker:Y:CENTer ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.SPAN

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.SPAN = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.SPAN

Description

This command sets/gets band marker Y span, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y span
Data Type	Double precision floating point type (Double)
Range	0 ~ 2T
Preset Value	2T
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.AM1.TRACe1.BDMarker.Y.SPAN = Var
Var = SCPI.CALCulate.AM1.TRACe1.BDMarker.Y.SPAN
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

AM Menu: **Marker Search** > **Band Marker Y** > **Span**

AM Menu: **Marker Function** > **Band Marker Y** > **Span**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:BDMarker:Y:SPAN {0 ~ 2T}

:CALCulate:AM[1-1]:TRACe[1-1]:BDMarker:Y:SPAN?

Query Response

{0 ~ 2T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:BDMarker:Y:SPAN 0"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:BDMarker:Y:SPAN ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.STARt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.STARt = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.STARt

Description

This command sets/gets band marker Y start, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y start
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	-1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.AM1.TRACe1.BDMarker.Y.STARt = Var
Var = SCPI.CALCulate.AM1.TRACe1.BDMarker.Y.STARt
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

AM Menu: **Marker Search** > **Band Marker Y** > **Start**

AM Menu: **Marker Function** > **Band Marker Y** > **Start**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:BDMarker:Y:STARt {-1T ~ 1T}

:CALCulate:AM[1-1]:TRACe[1-1]:BDMarker:Y:STARt?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:BDMarker:Y:STARt -10000000000"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:BDMarker:Y:STARt ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.STATe = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.STATe

Description

This command sets/gets band marker visible on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y state
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.AM1.TRACe1.BDMarker.Y.STATe = Var
Var = SCPI.CALCulate.AM1.TRACe1.BDMarker.Y.STATe
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

AM Menu: **Marker Search** > **Band Marker Y** > **Band Marker Y**

AM Menu: **Marker Function** > **Band Marker Y** > **Band Marker Y**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:BDMarker:Y:STATe {ON|OFF|1|0}

:CALCulate:AM[1-1]:TRACe[1-1]:BDMarker:Y:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:BDMarker:Y:STATe 1"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:BDMarker:Y:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.STOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.STOP = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.STOP

Description

This command sets/gets band marker Y stop, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y stop
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.AM1.TRACe1.BDMarker.Y.STOP = Var
Var = SCPI.CALCulate.AM1.TRACe1.BDMarker.Y.STOP
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.AM(Ch).TRACe(Tr).BDMarker.Y.STATe

Equivalent Key

AM Menu: **Marker Search** > **Band Marker Y** > **Stop**

AM Menu: **Marker Function** > **Band Marker Y** > **Stop**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:BDMarker:Y:STOP {-1T ~ 1T}

:CALCulate:AM[1-1]:TRACe[1-1]:BDMarker:Y:STOP?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:BDMarker:Y:STOP -10000000000"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:BDMarker:Y:STOP ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.COPY

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.COPY = *Value*

Description

This command copies data to user trace, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Copies data to user trace
Data Type	Long integer type (Long)
Range	1 ~ 8
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.AM1.TRACe1.DATA.COPY = Var
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:DATA:COPY {1 ~ 8}

Query Response

{1 ~ 8} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:DATA:COPY 1 "
20 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.FDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.FDATa = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.FDATa

Description

This command sets/gets formatted trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Formatted trace data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.AM1.TRACe1.DATA.FDATa = Var
Var = SCPI.CALCulate.AM1.TRACe1.DATA.FDATa
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:DATA:FDATa {1...1601}

:CALCulate:AM[1-1]:TRACe[1-1]:DATA:FDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:AM1:TRACe1:DATA:FDATa 1,1601"
30 OUTPUT 717;":CALCulate:AM1:TRACe1:DATA:FDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.FMEMory

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.FMEMory = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.FMEMory

Description

This command sets/gets formatted memory data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Formatted memory data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.AM1.TRACe1.DATA.FMEMory = Var
Var = SCPI.CALCulate.AM1.TRACe1.DATA.FMEMory
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:DATA:FMEMory {1...1601}

:CALCulate:AM[1-1]:TRACe[1-1]:DATA:FMEMory?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:AM1:TRACe1:DATA:FMEMory 1,1601"
30 OUTPUT 717;":CALCulate:AM1:TRACe1:DATA:FMEMory ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.PDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.PDATa = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.PDATa

Description

This command sets/gets dBc data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	dBC data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.AM1.TRACe1.DATA.PDATa = Var
Var = SCPI.CALCulate.AM1.TRACe1.DATA.PDATa
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:DATA:PDATa {1...1601}

:CALCulate:AM[1-1]:TRACe[1-1]:DATA:PDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:AM1:TRACe1:DATA:PDATa 1,1601"
30 OUTPUT 717;":CALCulate:AM1:TRACe1:DATA:PDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.PMEMory

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.PMEMory = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.PMEMory

Description

This command sets/gets dBc memory data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	dBc memory data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.AM1.TRACe1.DATA.PMEMory = Var
Var = SCPI.CALCulate.AM1.TRACe1.DATA.PMEMory
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:DATA:PMEMory {1...1601}

:CALCulate:AM[1-1]:TRACe[1-1]:DATA:PMEMory?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:AM1:TRACe1:DATA:PMEMory 1,1601"
30 OUTPUT 717;":CALCulate:AM1:TRACe1:DATA:PMEMory ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.SDATa

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.SDATa

Description

This command gets spurious data (0 or 1), for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
Var = SCPI.CALCulate.AM1.TRACe1.DATA.SDATa
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:DATA:SDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:AM1:TRACe1:DATA:SDATa ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.SMEMory

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.SMEMory

Description

This command gets spurious memory data (0 or 1), for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious memory data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
Var = SCPI.CALCulate.AM1.TRACe1.DATA.SMEMory
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:DATA:SMEMory?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:AM1:TRACe1:DATA:SMEMory ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.UDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.UDATa = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.UDATa

Description

This command sets/gets unformatted trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Unformatted trace data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.AM1.TRACe1.DATA.UDATa = Var
Var = SCPI.CALCulate.AM1.TRACe1.DATA.UDATa
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:DATA:UDATa {1...1601}

:CALCulate:AM[1-1]:TRACe[1-1]:DATA:UDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:AM1:TRACe1:DATA:UDATa 1,1601"
30 OUTPUT 717;":CALCulate:AM1:TRACe1:DATA:UDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.UMEMory

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.UMEMory = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.UMEMory

Description

This command sets/gets unformatted memory data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Unformatted memory data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.AM1.TRACe1.DATA.UMEMory = Var
Var = SCPI.CALCulate.AM1.TRACe1.DATA.UMEMory
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).DATA.UDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:DATA:UMEMory {1...1601}

:CALCulate:AM[1-1]:TRACe[1-1]:DATA:UMEMory?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:AM1:TRACe1:DATA:UMEMory 1,1601"
30 OUTPUT 717;":CALCulate:AM1:TRACe1:DATA:UMEMory ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).EQUation.TEXT

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).EQUation.TEXT = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).EQUation.TEXT

Description

This command sets/gets equation string, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Equation string
Data Type	Character string type (String)
Range	254 chars
Preset Value	""
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	170, "Invalid equation label" 171, "Invalid equation

	expression"

Examples

```
Dim Var as String
SCPI.CALCulate.AM1.TRACe1.EQUation.TEXT = Var
Var = SCPI.CALCulate.AM1.TRACe1.EQUation.TEXT
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:EQUation:TEXT

:CALCulate:AM[1-1]:TRACe[1-1]:EQUation:TEXT?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:EQUation:TEXT Sample_Text"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:EQUation:TEXT ?"
30 ENTER 717;A$
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.DOMain.X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.DOMain.X = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.DOMain.X

Description

This command sets/gets X analysis range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	X analysis range
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	BDMarker
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.AM1.TRACe1.FUNCtion.DOMain.X = Var
Var = SCPI.CALCulate.AM1.TRACe1.FUNCtion.DOMain.X
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.DOMain.X

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.DOMain.Y

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.STATistics.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.TYPE

Equivalent Key

AM Menu: **Marker Function** > **Analysis Range (X)**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:FUNCtion:DOMain:X {FRANge|BDMarker}

:CALCulate:AM[1-1]:TRACe[1-1]:FUNCtion:DOMain:X?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:FUNCtion:DOMain:X FRANge"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:FUNCtion:DOMain:X ?"
30 ENTER 717;A$
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.DOMain.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.DOMain.Y = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.DOMain.Y

Description

This command sets/gets Y analysis range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Y analysis range
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	BDMarker
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.AM1.TRACe1.FUNCtion.DOMain.Y = Var
Var = SCPI.CALCulate.AM1.TRACe1.FUNCtion.DOMain.Y
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.DOMain.X

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.STATistics.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.TYPE

Equivalent Key

AM Menu: **Marker Function** > **Analysis Range (Y)**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:FUNCtion:DOMain:Y {FRANge|BDMarker}

:CALCulate:AM[1-1]:TRACe[1-1]:FUNCtion:DOMain:Y?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:FUNCtion:DOMain:Y FRANge"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:FUNCtion:DOMain:Y ?"
30 ENTER 717;A$
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak

Description

This command gets trace data statistics, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*mean*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*std_dev*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*peak_to_peak*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-

Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Multiple
Var= FRANge
Var = SCPI.CALCulate.AM1.TRACe1.FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.DOMain.X

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.DOMain.Y

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.TYPE

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:FUNCtion:STATistics:DATA?

Query Response

{ - - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:FUNCtion:STATistics:DATA ?"
30 ENTER 717;A$
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak

Description

This command gets memory data statistics, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*mean*
Description	Memory data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*std_dev*
Description	Memory data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*peak_to_peak*
Description	Memory data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-

Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Multiple
Var= FRANge
Var = SCPI.CALCulate.AM1.TRACe1.FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.DOMain.X

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.DOMain.Y

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.STATistics.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.TYPE

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:FUNCtion:STATistics:MEMory?

Query Response

{ - - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:FUNCtion:STATistics:MEMory ?"
30 ENTER 717;A$
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.TYPE

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.TYPE = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.TYPE

Description

This command sets/gets Analysis type, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Analysis type
Data Type	Character string type (String)
Range	OFF|STATistics
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.AM1.TRACe1.FUNCtion.TYPE = Var
Var = SCPI.CALCulate.AM1.TRACe1.FUNCtion.TYPE
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.DOMain.X

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.DOMain.Y

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.STATistics.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory

Equivalent Key

AM Menu: **Marker Function** > **Analysis Type**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:FUNCtion:TYPE {OFF|STATistics}

:CALCulate:AM[1-1]:TRACe[1-1]:FUNCtion:TYPE?

Query Response

{OFF|STATistics} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:FUNCtion:TYPE OFF"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:FUNCtion:TYPE ?"
30 ENTER 717;A$
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).HOLD

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).HOLD = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).HOLD

Description

This command sets/gets Data hold, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Data hold
Data Type	Character string type (String)
Range	OFF|MAXimum|MINimum
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.AM1.TRACe1.HOLD = Var
Var = SCPI.CALCulate.AM1.TRACe1.HOLD
```

Equivalent Key

AM Menu: **Trace View** > **Data Hold**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:HOLD {OFF|MAXimum|MINimum}

:CALCulate:AM[1-1]:TRACe[1-1]:HOLD?

Query Response

{OFF|MAXimum|MINimum} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:HOLD OFF"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:HOLD ?"
30 ENTER 717;A$
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.FAIL

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.FAIL

Description

This command gets limit test pass/fail, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test result
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Boolean
Var= 1
Var = SCPI.CALCulate.AM1.TRACe1.LIMit.FAIL
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.REPort.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:LIMit:FAIL?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:LIMit:FAIL ?"
20 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.LDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.LDATa = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.LDATa

Description

This command sets/gets expanded lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Lower limit line
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.AM1.TRACe1.LIMit.LOWer.LDATa = Var
Var = SCPI.CALCulate.AM1.TRACe1.LIMit.LOWer.LDATa
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.FAIL

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.REPort.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:LIMit:LOWer:LDATa {1...1601}

:CALCulate:AM[1-1]:TRACe[1-1]:LIMit:LOWer:LDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:AM1:TRACe1:LIMit:LOWer:LDATa 1,1601"
30 OUTPUT 717;":CALCulate:AM1:TRACe1:LIMit:LOWer:LDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

Description

This command sets Clear lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.AM1.TRACe1.LIMit.LOWer.SEGMent.CLEar

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.FAIL

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.REPort.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:CLEar

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:LIMit:LOWer:SEGMent:CLEar"
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

Description

This command sets/gets number of lower limit segments, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Lower limit segments
Data Type	Long integer type (Long)
Range	1 ~ 100
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.AM1.TRACe1.LIMit.LOWer.SEGMent.COUNt = Var
Var = SCPI.CALCulate.AM1.TRACe1.LIMit.LOWer.SEGMent.COUNt
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.FAIL

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.REPort.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:COUNt {1 ~ 100}

:CALCulate:AM[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:COUNt?

Query Response

{1 ~ 100} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:LIMit:LOWer:SEGMent:COUNt 1 "
20 OUTPUT 717;":CALCulate:AM1:TRACe1:LIMit:LOWer:SEGMent:COUNt ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Description

This command sets/gets lower limit line data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Lower limit data
Data Type	Variant type Array (Range)
Range	1...400
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 400) as Variant
SCPI.CALCulate.AM1.TRACe1.LIMit.LOWer.SEGMent.DATA = Var
Var = SCPI.CALCulate.AM1.TRACe1.LIMit.LOWer.SEGMent.DATA
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.FAIL

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:DATA {1...400}

:CALCulate:AM[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:DATA?

Query Response

{1...400} <newline><^END>

Example of use

```
10 Dim A(1:400)
20 OUTPUT 717;":CALCulate:AM1:TRACe1:LIMit:LOWer:SEGMent:DATA 1,400"
30 OUTPUT 717;":CALCulate:AM1:TRACe1:LIMit:LOWer:SEGMent:DATA ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.REPort.DATA

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.REPort.DATA

Description

This command gets pass/fail list, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test report
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
Var = SCPI.CALCulate.AM1.TRACe1.LIMit.REPort.DATA
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.FAIL

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:LIMit:REPort[:DATA]?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:AM1:TRACe1:LIMit:REPort[:DATA] ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.STATe = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.STATe

Description

This command sets/gets limit test on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.AM1.TRACe1.LIMit.STATe = Var
Var = SCPI.CALCulate.AM1.TRACe1.LIMit.STATe
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.FAIL

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.REPort.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

AM Menu: **Display** > **Limit Test** > **Limit Test**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:LIMit[:STATe] {ON|OFF|1|0}

:CALCulate:AM[1-1]:TRACe[1-1]:LIMit[:STATe]?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:LIMit[:STATe] 1"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:LIMit[:STATe] ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.LDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.LDATa = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.LDATa

Description

This command sets/gets expanded upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Upper limit line
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.AM1.TRACe1.LIMit.UPPer.LDATa = Var
Var = SCPI.CALCulate.AM1.TRACe1.LIMit.UPPer.LDATa
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.FAIL

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.REPort.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:LIMit:UPPer:LDATa {1...1601}

:CALCulate:AM[1-1]:TRACe[1-1]:LIMit:UPPer:LDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:AM1:TRACe1:LIMit:UPPer:LDATa 1,1601"
30 OUTPUT 717;":CALCulate:AM1:TRACe1:LIMit:UPPer:LDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

Description

This command sets Clear upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.AM1.TRACe1.LIMit.UPPer.SEGMent.CLEar

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.FAIL

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.REPort.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:CLEar

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:LIMit:UPPer:SEGMent:CLEar"
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Description

This command sets/gets number of upper limit segments, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Upper limit segment
Data Type	Long integer type (Long)
Range	1 ~ 100
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.AM1.TRACe1.LIMit.UPPer.SEGMent.COUNt = Var
Var = SCPI.CALCulate.AM1.TRACe1.LIMit.UPPer.SEGMent.COUNt
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.FAIL

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.REPort.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:COUNt {1 ~ 100}

:CALCulate:AM[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:COUNt?

Query Response

{1 ~ 100} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:LIMit:UPPer:SEGMent:COUNt 1 "
20 OUTPUT 717;":CALCulate:AM1:TRACe1:LIMit:UPPer:SEGMent:COUNt ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Description

This command sets/gets upper limit line data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Upper limit line data
Data Type	Variant type Array (Range)
Range	1...400
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 400) as Variant
SCPI.CALCulate.AM1.TRACe1.LIMit.UPPer.SEGMent.DATA = Var
Var = SCPI.CALCulate.AM1.TRACe1.LIMit.UPPer.SEGMent.DATA
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.FAIL

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.REPort.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:DATA {1...400}

:CALCulate:AM[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:DATA?

Query Response

{1...400} <newline><^END>

Example of use

```
10 Dim A(1:400)
20 OUTPUT 717;":CALCulate:AM1:TRACe1:LIMit:UPPer:SEGMent:DATA 1,400"
30 OUTPUT 717;":CALCulate:AM1:TRACe1:LIMit:UPPer:SEGMent:DATA ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

Description

This command sets marker search LPEak, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.AM1.TRACe1.MARKer10.SEARch.EXECute.LPEak

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

AM Menu: **Marker Search** > **Peak** > **Search Left**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:LPEak

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:SEARch:EXECute:LPEak"
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

Description

This command sets marker search LSPurious, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

NOTE This function is allowed to use when the spurious display is power.

Examples

SCPI.CALCulate.AM1.TRACe1.MARKer10.SEARch.EXECute.LSPurious

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

AM Menu: **Marker Search** > **Spurious** > **Search Left**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute.LSPurious

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:SEARch:EXECute.LSPurious"
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

Description

This command sets marker search LTARget, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.AM1.TRACe1.MARKer10.SEARch.EXECute.LTARget

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

AM Menu: **Marker Search** > **Target** > **Search Left**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:LTARget

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:SEARch:EXECute:LTARget"
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

Description

This command sets marker search MAXimum, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.AM1.TRACe1.MARKer10.SEARch.EXECute.MAXimum

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

AM Menu: **Marker Search** > **Search Max**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:MAXimum

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:SEARch:EXECute:MAXimum"
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

Description

This command sets marker search MINimum, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.AM1.TRACe1.MARKer10.SEARch.EXECute.MINimum

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

AM Menu: **Marker Search** > **Search Min**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:MINimum

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:SEARch:EXECute:MINimum"
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

Description

This command sets marker search PEAK, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.AM1.TRACe1.MARKer10.SEARch.EXECute.PEAK

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

AM Menu: **Marker Search** > **Peak** > **Search Peak**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:PEAK

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:SEARch:EXECute:PEAK"
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

Description

This command sets marker search RPEak, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.AM1.TRACe1.MARKer10.SEARch.EXECute.RPEak

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

AM Menu: **Marker Search** > **Peak** > **Search Right**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:RPEak

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:SEARch:EXECute:RPEak"
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

Description

This command sets marker search RSPurious, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

NOTE This function is allowed to use when the spurious display is power.

Examples

SCPI.CALCulate.AM1.TRACe1.MARKer10.SEARch.EXECute.RSPurious

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

AM Menu: **Marker Search** > **Spurious** > **Search Right**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:RSPurious

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:SEARch:EXECute:RSPurious"
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

Description

This command sets marker search RTARget, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.AM1.TRACe1.MARKer10.SEARch.EXECute.RTARget

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

AM Menu: **Marker Search** > **Target** > **Search Right**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:RTARget

Example of use

10 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:SEARch:EXECute:RTARget"

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

Description

This command sets marker search SPURious, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

NOTE This function is allowed to use when the spurious display is power.

Examples

SCPI.CALCulate.AM1.TRACe1.MARKer10.SEARch.EXECute.SPURious

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

AM Menu: **Marker Search** > **Spurious** > **Search Spurious**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:SPURious

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:SEARch:EXECute:SPURious"
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Description

This command sets marker search TARGet, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.AM1.TRACe1.MARKer10.SEARch.EXECute.TARGet

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

Equivalent Key

AM Menu: **Marker Search** > **Target** > **Search Target**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:TARGet

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:SEARch:EXECute:TARGet"
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion

Description

This command sets/gets marker-search-peak excursion value, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search peak excursion value
Data Type	Double precision floating point type (Double)
Range	0 ~ 10G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.AM1.TRACe1.MARKer10.SEARch.PEAK.EXCursion = Var
Var = SCPI.CALCulate.AM1.TRACe1.MARKer10.SEARch.PEAK.EXCursion
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity

Equivalent Key

AM Menu: **Marker Search** > **Peak** > **Peak Excursion**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:PEAK:EXCursion {0 ~ 10G}

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:PEAK:EXCursion?

Query Response

{0 ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:SEARch:PEAK:EXCursion 0"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:SEARch:PEAK:EXCursion ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity

Description

This command sets/gets marker-search-peak polarity type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search peak polarity type
Data Type	Character string type (String)
Range	POSitive|NEGative|BOTH
Preset Value	POSitive
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "POSitive"
SCPI.CALCulate.AM1.TRACe1.MARKer10.SEARch.PEAK.POLarity = Var
Var = SCPI.CALCulate.AM1.TRACe1.MARKer10.SEARch.PEAK.POLarity
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion

Equivalent Key

AM Menu: **Marker Search** > **Peak** > **Peak Polarity**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:PEAK:POLarity {POSitive|NEGative|BOTH}

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:PEAK:POLarity?

Query Response

{POSitive|NEGative|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:SEARch:PEAK:POLarity POSitive"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:SEARch:PEAK:POLarity ?"
30 ENTER 717;A$
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition

Description

This command sets/gets marker-target transition type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker target transition type
Data Type	Character string type (String)
Range	POSitive|NEGative|BOTH
Preset Value	BOTH
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as String
Var= "POSitive"
SCPI.CALCulate.AM1.TRACe1.MARKer10.SEARch.TARGet.TRANsition = Var
Var = SCPI.CALCulate.AM1.TRACe1.MARKer10.SEARch.TARGet.TRANsition
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y

Equivalent Key

AM Menu: **Marker Search** > **Target** > **Target Transition**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TARGet:TRANsition {POSitive|NEGative|BOTH}

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TARGet:TRANsition?

Query Response

{POSitive|NEGative|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:SEARch:TARGet:TRANsition POSitive"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:SEARch:TARGet:TRANsition ?"
30 ENTER 717;A$
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y

Description

This command sets/gets marker target value, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker target value
Data Type	Double precision floating point type (Double)
Range	-10G ~ 10G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.AM1.TRACe1.MARKer10.SEARch.TARGet.Y = Var
Var = SCPI.CALCulate.AM1.TRACe1.MARKer10.SEARch.TARGet.Y
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition

Equivalent Key

AM Menu: **Marker Search** > **Target** > **Target Value**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TARGet:Y {-10G ~ 10G}

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TARGet:Y?

Query Response

{-10G ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:SEARch:TARGet:Y -10000000000"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:SEARch:TARGet:Y ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE

Description

This command sets/gets marker search tracking type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search tracking type
Data Type	Character string type (String)
Range	OFF|MAXimum|MINimum|PEAK|TARGet|SPURious
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.AM1.TRACe1.MARKer10.SEARch.TRACking.TYPE = Var
Var = SCPI.CALCulate.AM1.TRACe1.MARKer10.SEARch.TRACking.TYPE
```

Equivalent Key

AM Menu: **Marker Search** > **Tracking**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TRACking:TYPE {OFF|MAXimum|MINimum|PEAK|TARGet|SPURious}

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TRACking:TYPE?

Query Response

{OFF|MAXimum|MINimum|PEAK|TARGet|SPURious} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:SEARch:TRACking:TYPE OFF"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:SEARch:TRACking:TYPE ?"
30 ENTER 717;A$
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).STATe = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).STATe

Description

This command sets/gets marker visible on/off, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker visible state
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.AM1.TRACe1.MARKer10.STATe = Var
Var = SCPI.CALCulate.AM1.TRACe1.MARKer10.STATe
```

Equivalent Key

AM Menu: **Marker** > **Clear Marker Menu** > **Marker 1**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:STATe {ON|OFF|1|0}

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:STATe 1"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).X = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).X

Description

This command sets/gets marker x position., for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker X position
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	1k
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
SCPI.CALCulate.AM1.TRACe1.MARKer10.X = Var
Var = SCPI.CALCulate.AM1.TRACe1.MARKer10.X
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).Y

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:X

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:X?

Query Response

{Double} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:X 0"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:X ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).Y

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).Y

Description

This command gets marker y position, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker Y position
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var = SCPI.CALCulate.AM1.TRACe1.MARKer10.Y
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).MARKer(Mk).X

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:MARKer[1-10]:Y?

Query Response

{Double} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:MARKer10:Y ?"
20 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).MATH.FUNCtion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).MATH.FUNCtion = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).MATH.FUNCtion

Description

This command sets/gets math operation type, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Math function operation
Data Type	Character string type (String)
Range	NORMal|SUBTract|DIVide|ADD|MULTiply|USER
Preset Value	NORMal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NORMal"
SCPI.CALCulate.AM1.TRACe1.MATH.FUNCtion = Var
Var = SCPI.CALCulate.AM1.TRACe1.MATH.FUNCtion
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).MATH.MEMorize

SCPI.CALCulate.AM(Ch).TRACe(Tr).MATH.OFFSet

Equivalent Key

AM Menu: **Trace View** > **Data Math**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:MATH:FUNCtion
{NORMal|SUBTract|DIVide|ADD|MULTiply|USER}

:CALCulate:AM[1-1]:TRACe[1-1]:MATH:FUNCtion?

Query Response

{NORMal|SUBTract|DIVide|ADD|MULTiply|USER} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:MATH:FUNCtion NORMal"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:MATH:FUNCtion ?"
30 ENTER 717;A$
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).MATH.MEMorize

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).MATH.MEMorize

Description

This command copies data to memory, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.AM1.TRACe1.MATH.MEMorize

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).MATH.FUNCtion

SCPI.CALCulate.AM(Ch).TRACe(Tr).MATH.OFFSet

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:MATH:MEMorize

Example of use

10 OUTPUT 717;":CALCulate:AM1:TRACe1:MATH:MEMorize"

SCPI.CALCulate.AM(Ch).TRACe(Tr).MATH.OFFSet

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).MATH.OFFSet = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).MATH.OFFSet

Description

This command sets/gets Trace Offset, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace offset
Data Type	Double precision floating point type (Double)
Range	-500G ~ 500G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -500000000000
SCPI.CALCulate.AM1.TRACe1.MATH.OFFSet = Var
Var = SCPI.CALCulate.AM1.TRACe1.MATH.OFFSet
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).MATH.FUNCtion

SCPI.CALCulate.AM(Ch).TRACe(Tr).MATH.MEMorize

Equivalent Key

AM Menu: **Trace View** > **Offset**

AM Menu: **Trace View** > **Marker** > **-Offset**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:MATH:OFFSet {-500G ~ 500G}

:CALCulate:AM[1-1]:TRACe[1-1]:MATH:OFFSet?

Query Response

{-500G ~ 500G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:MATH:OFFSet -500000000000"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:MATH:OFFSet ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.APERture

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.APERture = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.APERture

Description

This command sets/gets Smoothing aperture, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Smoothing aperture
Data Type	Double precision floating point type (Double)
Range	50m ~ 25
Preset Value	1.5
Unit	%
Resolution	10m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0.05
SCPI.CALCulate.AM1.TRACe1.SMOothing.APERture = Var
Var = SCPI.CALCulate.AM1.TRACe1.SMOothing.APERture
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.APERture

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SENSibility

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

AM Menu: **Trace View** > **Aperture**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:SMOothing:APERture {50m ~ 25}

:CALCulate:AM[1-1]:TRACe[1-1]:SMOothing:APERture?

Query Response

{50m ~ 25} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:SMOothing:APERture 0.05"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:SMOothing:APERture ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.STATe = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.STATe

Description

This command sets/gets Smoothing on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Smoothing (On/OFF)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.AM1.TRACe1.SMOothing.STATe = Var
Var = SCPI.CALCulate.AM1.TRACe1.SMOothing.STATe
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.APERture

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SENSibility

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

AM Menu: **Trace View** > **Smoothing**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:SMOothing:STATe {ON|OFF|1|0}

:CALCulate:AM[1-1]:TRACe[1-1]:SMOothing:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:SMOothing:STATe 1"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:SMOothing:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OMISsion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OMISsion = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OMISsion

Description

This command sets/gets Spurious display omission ON/OFF, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious display omission
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.AM1.TRACe1.SPURious.OMISsion = Var
Var = SCPI.CALCulate.AM1.TRACe1.SPURious.OMISsion
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.APERture

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:SPURious:OMISsion {ON|OFF|1|0}

:CALCulate:AM[1-1]:TRACe[1-1]:SPURious:OMISsion?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:SPURious:OMISsion 1"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:SPURious:OMISsion ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

Description

This command clears threshold table, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.AM1.TRACe1.SPURious.OSSPur.FLISt.CLEar

Related Objects

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:SPURious:OSSPur:FLISt:CLEar

Example of use

10 OUTPUT 717;":CALCulate:AM1:TRACe1:SPURious:OSSPur:FLISt:CLEar"

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

Description

This command sets/gets Spurious Table for Omit Specified Spurious, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious table for omit specified spurious
Data Type	Long integer type (Long)
Range	1 ~ 100
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.AM1.TRACe1.SPURious.OSSPur.FLISt.COUNt = Var
Var = SCPI.CALCulate.AM1.TRACe1.SPURious.OSSPur.FLISt.COUNt
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.APERture

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SENSibility

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:SPURious:OSSPur:FLISt:COUNt {1 ~ 100}

:CALCulate:AM[1-1]:TRACe[1-1]:SPURious:OSSPur:FLISt:COUNt?

Query Response

{1 ~ 100} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:SPURious:OSSPur:FLISt:COUNt 1 "
20 OUTPUT 717;":CALCulate:AM1:TRACe1:SPURious:OSSPur:FLISt:COUNt ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

Description

This command sets/gets threshold table, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Threshold table
Data Type	Variant type Array (Range)
Range	1...100
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 100) as Variant
SCPI.CALCulate.AM1.TRACe1.SPURious.OSSPur.FLISt.DATA = Var
Var = SCPI.CALCulate.AM1.TRACe1.SPURious.OSSPur.FLISt.DATA
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.APERture

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SENSibility

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:SPURious:OSSPur:FLISt:DATA {1...100}

:CALCulate:AM[1-1]:TRACe[1-1]:SPURious:OSSPur:FLISt:DATA?

Query Response

{1...100} <newline><^END>

Example of use

```
10 Dim A(1:100)
20 OUTPUT 717;":CALCulate:AM1:TRACe1:SPURious:OSSPur:FLISt:DATA 1,100"
```

```
30 OUTPUT 717;":CALCulate:AM1:TRACe1:SPURious:OSSPur:FLISt:DATA ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

Description

This command sets/gets Specified Spurious display omission ON/OFF, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious display mission for the selected trace
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.AM1.TRACe1.SPURious.OSSPur.OMISsion = Var
Var = SCPI.CALCulate.AM1.TRACe1.SPURious.OSSPur.OMISsion
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.APERture

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SENSibility

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:SPURious:OSSPur:OMISsion {ON|OFF|1|0}

:CALCulate:AM[1-1]:TRACe[1-1]:SPURious:OSSPur:OMISsion?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:SPURious:OSSPur:OMISsion 1"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:SPURious:OSSPur:OMISsion ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.POWer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.POWer = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.POWer

Description

This command sets/gets Spurious display merging ON/OFF, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious display merging
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.AM1.TRACe1.SPURious.POWer = Var
Var = SCPI.CALCulate.AM1.TRACe1.SPURious.POWer
```

Related Objects

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:SPURious:POWer {ON|OFF|1|0}

:CALCulate:AM[1-1]:TRACe[1-1]:SPURious:POWer?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:SPURious:POWer 1"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:SPURious:POWer ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SENSibility

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SENSibility = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SENSibility

Description

This command sets/gets spurious sensibility, for the selected trace *Tr* of the selected channel *Ch*.

The low peak has real sensibility when the value is small.

Variable

Parameter	*Value*
Description	Spurious sensibility
Data Type	Double precision floating point type (Double)
Range	10m ~ 10
Preset Value	3
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 5
SCPI.CALCulate.AM1.TRACe1.SPURious.SENSibility = Var
Var = SCPI.CALCulate.AM1.TRACe1.SPURious.SENSibility
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.APERture

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

AM Menu: **Trace View** > **Spurious** > **Spur Sensibility**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:SPURious:SENSibility {10m ~ 10}

:CALCulate:AM[1-1]:TRACe[1-1]:SPURious:SENSibility?

Query Response

{10m ~ 10} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:SPURious:SENSibility 5"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:SPURious:SENSibility?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SLISt

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SLISt

Description

This command gets Spurious List, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious list
Data Type	Variant type Array (Range)
Range	1...0
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 0) as Variant
Var = SCPI.CALCulate.AM1.TRACe1.SPURious.SLISt
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.APERture

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SENSibility

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:SPURious:SLISt?

Query Response

{1...0} <newline><^END>

Example of use

```
10 Dim A(1:0)
20 OUTPUT 717;":CALCulate:AM1:TRACe1:SPURious:SLISt ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

Description

This command sets/gets spurious minimum level definition, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious minimum level definition
Data Type	Double precision floating point type (Double)
Range	-500 ~ 500
Preset Value	-500
Unit	dBc
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.AM1.TRACe1.SPURious.THReshold.LEVel.MINimum = Var
Var = SCPI.CALCulate.AM1.TRACe1.SPURious.THReshold.LEVel.MINimum
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.APERture

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SENSibility

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

AM Menu: **Trace View** > **Spurious** > **Minimum Spur Level**

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:SPURious:THReshold:LEVel:MINimum {-500 ~ 500}

:CALCulate:AM[1-1]:TRACe[1-1]:SPURious:THReshold:LEVel:MINimum?

Query Response

{-500 ~ 500} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:SPURious:THReshold:LEVel:MINimum 0"
20 OUTPUT 717;":CALCulate:AM1:TRACe1:SPURious:THReshold:LEVel:MINimum ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

Description

This command clears threshold table, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.AM1.TRACe1.SPURious.THReshold.TABLe.CLEar

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.APERture

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SENSibility

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:CLEar

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:SPURious:THReshold:TABLe:CLEar"
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

Description

This command sets/gets number of threshold segments, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Number of threshold segments
Data Type	Long integer type (Long)
Range	1 ~ 20
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.AM1.TRACe1.SPURious.THReshold.TABLe.COUNt = Var
Var = SCPI.CALCulate.AM1.TRACe1.SPURious.THReshold.TABLe.COUNt
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.APERture

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SENSibility

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:COUNt {1 ~ 20}

:CALCulate:AM[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:COUNt?

Query Response

{1 ~ 20} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:AM1:TRACe1:SPURious:THReshold:TABLe:COUNt 1 "
20 OUTPUT 717;":CALCulate:AM1:TRACe1:SPURious:THReshold:TABLe:COUNt ?"
30 ENTER 717;A
```

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA = *Value*

Value = SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Description

This command sets/gets threshold table, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Threshold table
Data Type	Variant type Array (Range)
Range	1...60
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 60) as Variant
SCPI.CALCulate.AM1.TRACe1.SPURious.THReshold.TABLe.DATA = Var
Var = SCPI.CALCulate.AM1.TRACe1.SPURious.THReshold.TABLe.DATA
```

Related Objects

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.APERture

SCPI.CALCulate.AM(Ch).TRACe(Tr).SMOothing.STATe

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SENSibility

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.AM(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:AM[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:DATA {1...60}

:CALCulate:AM[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:DATA?

Query Response

{1...60} <newline><^END>

Example of use

```
10 Dim A(1:60)
20 OUTPUT 717;":CALCulate:AM1:TRACe1:SPURious:THReshold:TABLe:DATA 1,60"
```

```
30 OUTPUT 717;":CALCulate:AM1:TRACe1:SPURious:THReshold:TABLe:DATA ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.BB(Ch).ALLTrace.LIMit.FAIL

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.BB(Ch).ALLTrace.LIMit.FAIL

Description

This command gets limit test pass/fail, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Result of limit test
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Boolean
Var= 1
Var = SCPI.CALCulate.BB1.ALLTrace.LIMit.FAIL
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:ALLTrace:LIMit:FAIL?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:ALLTrace:LIMit:FAIL ?"
20 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.COUPle.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.COUPle.STATe = *Value*

Value = SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.COUPle.STATe

Description

This command sets/gets marker couple on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker Coupling Function state (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.BB1.ALLTrace.MARKer.COUPle.STATe = Var
Var = SCPI.CALCulate.BB1.ALLTrace.MARKer.COUPle.STATe
```

Related Objects

SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.DISCrete.STATe

SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.REFerence.NUMBer

SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.REFerence.STATe

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:ALLTrace:MARKer:COUPle:STATe {ON|OFF|1|0}

:CALCulate:BB[1-1]:ALLTrace:MARKer:COUPle:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:ALLTrace:MARKer:COUPle:STATe 1"
20 OUTPUT 717;":CALCulate:BB1:ALLTrace:MARKer:COUPle:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.DISCrete.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.DISCrete.STATe = *Value*

Value = SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.DISCrete.STATe

Description

This command sets/gets marker discrete on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Enables/Disables marker discrete
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.BB1.ALLTrace.MARKer.DISCrete.STATe = Var
Var = SCPI.CALCulate.BB1.ALLTrace.MARKer.DISCrete.STATe
```

Related Objects

SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.COUPle.STATe

SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.REFerence.NUMBer

SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.REFerence.STATe

Equivalent Key

BB Menu: **Marker** > **More Functions** > **Discrete**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:ALLTrace:MARKer:DISCrete:STATe {ON|OFF|1|0}

:CALCulate:BB[1-1]:ALLTrace:MARKer:DISCrete:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:ALLTrace:MARKer:DISCrete:STATe 1"
20 OUTPUT 717;":CALCulate:BB1:ALLTrace:MARKer:DISCrete:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.REFerence.NUMBer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.REFerence.NUMBer = *Value*

Value = SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.REFerence.NUMBer

Description

This command sets/gets reference marker number, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker reference number
Data Type	Long integer type (Long)
Range	1 ~ 10
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.BB1.ALLTrace.MARKer.REFerence.NUMBer = Var
Var = SCPI.CALCulate.BB1.ALLTrace.MARKer.REFerence.NUMBer
```

Related Objects

SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.COUPle.STATe

SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.DISCrete.STATe

SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.REFerence.STATe

Equivalent Key

BB Menu: **Marker** > **More Functions** > **Ref Marker**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:ALLTrace:MARKer:REFerence:NUMBer {1 ~ 10}

:CALCulate:BB[1-1]:ALLTrace:MARKer:REFerence:NUMBer?

Query Response

{1 ~ 10} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:ALLTrace:MARKer:REFerence:NUMBer 1 "
20 OUTPUT 717;":CALCulate:BB1:ALLTrace:MARKer:REFerence:NUMBer ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.REFerence.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.REFerence.STATe = *Value*

Value = SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.REFerence.STATe

Description

This command sets/gets delta marker mode on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Delta marker mode (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.BB1.ALLTrace.MARKer.REFerence.STATe = Var
Var = SCPI.CALCulate.BB1.ALLTrace.MARKer.REFerence.STATe
```

Related Objects

SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.COUPle.STATe

SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.DISCrete.STATe

SCPI.CALCulate.BB(Ch).ALLTrace.MARKer.REFerence.NUMBer

Equivalent Key

BB Menu: **Marker** > **More Functions** > **Ref Marker Mode**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:ALLTrace:MARKer:REFerence:STATe {ON|OFF|1|0}

:CALCulate:BB[1-1]:ALLTrace:MARKer:REFerence:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:ALLTrace:MARKer:REFerence:STATe 1"
20 OUTPUT 717;":CALCulate:BB1:ALLTrace:MARKer:REFerence:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).DATA.PDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).DATA.PDATa = *Value*

Value = SCPI.CALCulate.BB(Ch).DATA.PDATa

Description

This command sets/gets measurement raw power data, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Measurement raw power data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.BB1.DATA.PDATa = Var
Var = SCPI.CALCulate.BB1.DATA.PDATa
```

Related Objects

SCPI.CALCulate.BB(Ch).DATA.RDATa

SCPI.CALCulate.BB(Ch).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:DATA:PDATa {1...1601}

:CALCulate:BB[1-1]:DATA:PDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:BB1:DATA:PDATa 1,1601"
30 OUTPUT 717;":CALCulate:BB1:DATA:PDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.BB(Ch).DATA.RDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).DATA.RDATa = *Value*

Value = SCPI.CALCulate.BB(Ch).DATA.RDATa

Description

This command sets/gets measurement raw data, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Measurement raw power data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.BB1.DATA.RDATa = Var
Var = SCPI.CALCulate.BB1.DATA.RDATa
```

Related Objects

SCPI.CALCulate.BB(Ch).DATA.PDATa

SCPI.CALCulate.BB(Ch).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:DATA:RDATa {1...1601}

:CALCulate:BB[1-1]:DATA:RDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:BB1:DATA:RDATa 1,1601"
30 OUTPUT 717;":CALCulate:BB1:DATA:RDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.BB(Ch).DATA.XDATa

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.BB(Ch).DATA.XDATa

Description

This command gets X axis data, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	X axis data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
Var = SCPI.CALCulate.BB1.DATA.XDATa
```

Related Objects

SCPI.CALCulate.BB(Ch).DATA.PDATa

SCPI.CALCulate.BB(Ch).DATA.RDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:DATA:XDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:BB1:DATA:XDATa ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).ALLMarker.ACTive

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).ALLMarker.ACTive = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).ALLMarker.ACTive

Description

This command sets/gets active marker, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Active marker
Data Type	Long integer type (Long)
Range	1 ~ 10
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.BB1.TRACe1.ALLMarker.ACTive = Var
Var = SCPI.CALCulate.BB1.TRACe1.ALLMarker.ACTive
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:ALLMarker:ACTive {1 ~ 10}

:CALCulate:BB[1-1]:TRACe[1-1]:ALLMarker:ACTive?

Query Response

{1 ~ 10} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:ALLMarker:ACTive 1 "
20 OUTPUT 717;":CALCulate:BB1:TRACe1:ALLMarker:ACTive ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

Description

This command sets/gets marker search X range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search X range source
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	FRANge
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.BB1.TRACe1.ALLMarker.SEARch.DOMain.X = Var
Var = SCPI.CALCulate.BB1.TRACe1.ALLMarker.SEARch.DOMain.X
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

SCPI.CALCulate.BB(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Equivalent Key

BB Menu: **Marker Search** > **Search Range (X)**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:ALLMarker:SEARch:DOMain:X {FRANge|BDMarker}

:CALCulate:BB[1-1]:TRACe[1-1]:ALLMarker:SEARch:DOMain:X?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:ALLMarker:SEARch:DOMain:X FRANge"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:ALLMarker:SEARch:DOMain:X ?"
30 ENTER 717;A$
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Description

This command sets/gets marker search Y range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search Y range
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	FRANge
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.BB1.TRACe1.ALLMarker.SEARch.DOMain.Y = Var
Var = SCPI.CALCulate.BB1.TRACe1.ALLMarker.SEARch.DOMain.Y
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

SCPI.CALCulate.BB(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Equivalent Key

BB Menu: **Marker Search** > **Search Range (Y)**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:ALLMarker:SEARch:DOMain:Y {FRANge|BDMarker}

:CALCulate:BB[1-1]:TRACe[1-1]:ALLMarker:SEARch:DOMain:Y?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:ALLMarker:SEARch:DOMain:Y FRANge"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:ALLMarker:SEARch:DOMain:Y ?"
30 ENTER 717;A$
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Description

This command sets search peak all, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.BB1.TRACe1.ALLMarker.SEARch.PEAK

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

SCPI.CALCulate.BB(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Equivalent Key

BB Menu: **Marker Search** > **Peak** > **Search Peak All**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:ALLMarker:SEARch:PEAK

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:ALLMarker:SEARch:PEAK"
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).ALLMarker.SEARch.SPURious

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).ALLMarker.SEARch.SPURious

Description

This command sets search spurious all, for the selected trace *Tr* of the selected channel *Ch*. You can display up to 10 markers.

NOTE This function is allowed to use when the spurious display is power.

Examples

SCPI.CALCulate.AM1.TRACe1.ALLMarker.SEARch.SPURious

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

Equivalent Key

BB Menu: **Marker Search** > **Spurious** > **Search Spurious All**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:ALLMarker:SEARch:SPURious

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:ALLMarker:SEARch:SPURious"
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.CENTer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.CENTer = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.CENTer

Description

This command sets/gets band marker X center, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X center
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.BB1.TRACe1.BDMarker.X.CENTer = Var
Var = SCPI.CALCulate.BB1.TRACe1.BDMarker.X.CENTer
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

BB Menu: **Marker Search** > **Band Marker X** > **Center**

BB Menu: **Marker Function** > **Band Marker X** > **Center**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:BDMarker:X:CENTer {-1T ~ 1T}

:CALCulate:BB[1-1]:TRACe[1-1]:BDMarker:X:CENTer?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:BDMarker:X:CENTer -10000000000"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:BDMarker:X:CENTer ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.SPAN

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.SPAN = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.SPAN

Description

This command sets/gets band marker X span, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X span
Data Type	Double precision floating point type (Double)
Range	0 ~ 2T
Preset Value	2T
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.BB1.TRACe1.BDMarker.X.SPAN = Var
Var = SCPI.CALCulate.BB1.TRACe1.BDMarker.X.SPAN
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

BB Menu: **Marker Search** > **Band Marker X** > **Span**

BB Menu: **Marker Function** > **Band Marker X** > **Span**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:BDMarker:X:SPAN {0 ~ 2T}

:CALCulate:BB[1-1]:TRACe[1-1]:BDMarker:X:SPAN?

Query Response

{0 ~ 2T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:BDMarker:X:SPAN 0"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:BDMarker:X:SPAN ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.STARt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.STARt = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.STARt

Description

This command sets/gets band marker X start, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X start
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	-1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.BB1.TRACe1.BDMarker.X.STARt = Var
Var = SCPI.CALCulate.BB1.TRACe1.BDMarker.X.STARt
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

BB Menu: **Marker Search** > **Band Marker X** > **Start**

BB Menu: **Marker Function** > **Band Marker X** > **Start**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:BDMarker:X:STARt {-1T ~ 1T}

:CALCulate:BB[1-1]:TRACe[1-1]:BDMarker:X:STARt?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:BDMarker:X:STARt -10000000000"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:BDMarker:X:STARt ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.STATe = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.STATe

Description

This command sets/gets band marker visible on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker visibility (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.BB1.TRACe1.BDMarker.X.STATe = Var
Var = SCPI.CALCulate.BB1.TRACe1.BDMarker.X.STATe
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

BB Menu: **Marker Search** > **Band Marker X** > **Band Marker X**

BB Menu: **Marker Function** > **Band Marker X** > **Band Marker X**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:BDMarker:X:STATe {ON|OFF|1|0}

:CALCulate:BB[1-1]:TRACe[1-1]:BDMarker:X:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:BDMarker:X:STATe 1"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:BDMarker:X:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.STOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.STOP = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.STOP

Description

This command sets/gets band marker X stop, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X stop
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.BB1.TRACe1.BDMarker.X.STOP = Var
Var = SCPI.CALCulate.BB1.TRACe1.BDMarker.X.STOP
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.X.STATe

Equivalent Key

BB Menu: **Marker Search** > **Band Marker X** > **Stop**

BB Menu: **Marker Function** > **Band Marker X** > **Stop**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:BDMarker:X:STOP {-1T ~ 1T}

:CALCulate:BB[1-1]:TRACe[1-1]:BDMarker:X:STOP?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:BDMarker:X:STOP -10000000000"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:BDMarker:X:STOP ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.CENTer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.CENTer = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.CENTer

Description

This command sets/gets band marker Y center, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y center
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.BB1.TRACe1.BDMarker.Y.CENTer = Var
Var = SCPI.CALCulate.BB1.TRACe1.BDMarker.Y.CENTer
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

BB Menu: **Marker Search** > **Band Marker Y** > **Center**

BB Menu: **Marker Function** > **Band Marker Y** > **Center**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:BDMarker:Y:CENTer {-1T ~ 1T}

:CALCulate:BB[1-1]:TRACe[1-1]:BDMarker:Y:CENTer?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:BDMarker:Y:CENTer -10000000000"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:BDMarker:Y:CENTer ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.SPAN

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.SPAN = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.SPAN

Description

This command sets/gets band marker Y span, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y span
Data Type	Double precision floating point type (Double)
Range	0 ~ 2T
Preset Value	2T
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.BB1.TRACe1.BDMarker.Y.SPAN = Var
Var = SCPI.CALCulate.BB1.TRACe1.BDMarker.Y.SPAN
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

BB Menu: **Marker Search** > **Band Marker Y** > **Span**

BB Menu: **Marker Function** > **Band Marker Y** > **Span**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:BDMarker:Y:SPAN {0 ~ 2T}

:CALCulate:BB[1-1]:TRACe[1-1]:BDMarker:Y:SPAN?

Query Response

{0 ~ 2T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:BDMarker:Y:SPAN 0"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:BDMarker:Y:SPAN ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.STARt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.STARt = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.STARt

Description

This command sets/gets band marker Y start, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y start
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	-1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.BB1.TRACe1.BDMarker.Y.STARt = Var
Var = SCPI.CALCulate.BB1.TRACe1.BDMarker.Y.STARt
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

BB Menu: **Marker Search** > **Band Marker Y** > **Start**

BB Menu: **Marker Function** > **Band Marker Y** > **Start**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:BDMarker:Y:STARt {-1T ~ 1T}

:CALCulate:BB[1-1]:TRACe[1-1]:BDMarker:Y:STARt?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:BDMarker:Y:STARt -10000000000"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:BDMarker:Y:STARt ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.STATe = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.STATe

Description

This command sets/gets band marker visible on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y state
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.BB1.TRACe1.BDMarker.Y.STATe = Var
Var = SCPI.CALCulate.BB1.TRACe1.BDMarker.Y.STATe
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

BB Menu: **Marker Search** > **Band Marker Y** > **Band Marker Y**

BB Menu: **Marker Function** > **Band Marker Y** > **Band Marker Y**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:BDMarker:Y:STATe {ON|OFF|1|0}

:CALCulate:BB[1-1]:TRACe[1-1]:BDMarker:Y:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:BDMarker:Y:STATe 1"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:BDMarker:Y:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.STOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.STOP = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.STOP

Description

This command sets/gets band marker Y stop, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y stop
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.BB1.TRACe1.BDMarker.Y.STOP = Var
Var = SCPI.CALCulate.BB1.TRACe1.BDMarker.Y.STOP
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.BB(Ch).TRACe(Tr).BDMarker.Y.STATe

Equivalent Key

BB Menu: **Marker Search** > **Band Marker Y** > **Stop**

BB Menu: **Marker Function** > **Band Marker Y** > **Stop**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:BDMarker:Y:STOP {-1T ~ 1T}

:CALCulate:BB[1-1]:TRACe[1-1]:BDMarker:Y:STOP?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:BDMarker:Y:STOP -10000000000"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:BDMarker:Y:STOP ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.COPY

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.COPY = *Value*

Description

This command sets copy to user trace, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Copies data to user trace
Data Type	Long integer type (Long)
Range	1 ~ 8
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.BB1.TRACe1.DATA.COPY = Var
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:DATA:COPY {1 ~ 8}

Query Response

{1 ~ 8} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:DATA:COPY 1 "
20 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.FDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.FDATa = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.FDATa

Description

This command sets/gets formatted trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Formatted trace data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.BB1.TRACe1.DATA.FDATa = Var
Var = SCPI.CALCulate.BB1.TRACe1.DATA.FDATa
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:DATA:FDATa {1...1601}

:CALCulate:BB[1-1]:TRACe[1-1]:DATA:FDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:BB1:TRACe1:DATA:FDATa 1,1601"
30 OUTPUT 717;":CALCulate:BB1:TRACe1:DATA:FDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.FMEMory

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.FMEMory = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.FMEMory

Description

This command sets/gets formatted memory data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Formatted memory data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.BB1.TRACe1.DATA.FMEMory = Var
Var = SCPI.CALCulate.BB1.TRACe1.DATA.FMEMory
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:DATA:FMEMory {1...1601}

:CALCulate:BB[1-1]:TRACe[1-1]:DATA:FMEMory?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:BB1:TRACe1:DATA:FMEMory 1,1601"
30 OUTPUT 717;":CALCulate:BB1:TRACe1:DATA:FMEMory ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.PDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.PDATa = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.PDATa

Description

This command sets/gets dBV data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	dBC data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.BB1.TRACe1.DATA.PDATa = Var
Var = SCPI.CALCulate.BB1.TRACe1.DATA.PDATa
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:DATA:PDATa {1...1601}

:CALCulate:BB[1-1]:TRACe[1-1]:DATA:PDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:BB1:TRACe1:DATA:PDATa 1,1601"
30 OUTPUT 717;":CALCulate:BB1:TRACe1:DATA:PDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.PMEMory

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.PMEMory = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.PMEMory

Description

This command sets/gets dBV memory data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	dBc memory data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.BB1.TRACe1.DATA.PMEMory = Var
Var = SCPI.CALCulate.BB1.TRACe1.DATA.PMEMory
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:DATA:PMEMory {1...1601}

:CALCulate:BB[1-1]:TRACe[1-1]:DATA:PMEMory?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:BB1:TRACe1:DATA:PMEMory 1,1601"
30 OUTPUT 717;":CALCulate:BB1:TRACe1:DATA:PMEMory ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.SDATa

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.SDATa

Description

This command gets spurious data (0 or 1), for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
Var = SCPI.CALCulate.BB1.TRACe1.DATA.SDATa
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:DATA:SDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:BB1:TRACe1:DATA:SDATa ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.SMEMory

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.SMEMory

Description

This command gets spurious memory data (0 or 1), for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious memory data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
Var = SCPI.CALCulate.BB1.TRACe1.DATA.SMEMory
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:DATA:SMEMory?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:BB1:TRACe1:DATA:SMEMory ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.UDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.UDATa = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.UDATa

Description

This command sets/gets unformatted trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Unformatted trace data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.BB1.TRACe1.DATA.UDATa = Var
Var = SCPI.CALCulate.BB1.TRACe1.DATA.UDATa
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:DATA:UDATa {1...1601}

:CALCulate:BB[1-1]:TRACe[1-1]:DATA:UDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:BB1:TRACe1:DATA:UDATa 1,1601"
30 OUTPUT 717;":CALCulate:BB1:TRACe1:DATA:UDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.UMEMory

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.UMEMory = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.UMEMory

Description

This command sets/gets unformatted memory data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Unformatted memory data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.BB1.TRACe1.DATA.UMEMory = Var
Var = SCPI.CALCulate.BB1.TRACe1.DATA.UMEMory
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).DATA.UDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:DATA:UMEMory {1...1601}

:CALCulate:BB[1-1]:TRACe[1-1]:DATA:UMEMory?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:BB1:TRACe1:DATA:UMEMory 1,1601"
30 OUTPUT 717;":CALCulate:BB1:TRACe1:DATA:UMEMory ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).EQUation.TEXT

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).EQUation.TEXT = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).EQUation.TEXT

Description

This command sets/gets equation string, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Equation string
Data Type	Character string type (String)
Range	254 chars
Preset Value	""
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	170, "Invalid equation label" 171, "Invalid equation

	expression"

Examples

```
Dim Var as String
SCPI.CALCulate.BB1.TRACe1.EQUation.TEXT = Var
Var = SCPI.CALCulate.BB1.TRACe1.EQUation.TEXT
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:EQUation:TEXT

:CALCulate:BB[1-1]:TRACe[1-1]:EQUation:TEXT?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:EQUation:TEXT Sample_Text"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:EQUation:TEXT ?"
30 ENTER 717;A$
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).FORMat

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).FORMat = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).FORMat

Description

This command sets/gets BB format, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	BB format
Data Type	Character string type (String)
Range	DBVHz|DBMHz|VHz
Preset Value	DBVHz
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "DBVHz"
SCPI.CALCulate.BB1.TRACe1.FORMat = Var
Var = SCPI.CALCulate.BB1.TRACe1.FORMat
```

Equivalent Key

BB Menu: **Format** > **Format**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:FORMat {DBVHz|DBMHz|VHz}

:CALCulate:BB[1-1]:TRACe[1-1]:FORMat?

Query Response

{DBVHz|DBMHz|VHz} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:FORMat DBVHz"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:FORMat ?"
30 ENTER 717;A$
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.DOMain.X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.DOMain.X = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.DOMain.X

Description

This command sets/gets X analysis range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	X analysis range
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	BDMarker
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.BB1.TRACe1.FUNCtion.DOMain.X = Var
Var = SCPI.CALCulate.BB1.TRACe1.FUNCtion.DOMain.X
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.DOMain.X

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.DOMain.Y

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.STATistics.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.TYPE

Equivalent Key

BB Menu: **Marker Function** > **Analysis Range (X)**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:FUNCtion:DOMain:X {FRANge|BDMarker}

:CALCulate:BB[1-1]:TRACe[1-1]:FUNCtion:DOMain:X?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:FUNCtion:DOMain:X FRANge"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:FUNCtion:DOMain:X ?"
30 ENTER 717;A$
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.DOMain.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.DOMain.Y = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.DOMain.Y

Description

This command sets/gets Y analysis range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Y analysis range source
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	BDMarker
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.BB1.TRACe1.FUNCtion.DOMain.Y = Var
Var = SCPI.CALCulate.BB1.TRACe1.FUNCtion.DOMain.Y
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.DOMain.X

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.STATistics.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.TYPE

Equivalent Key

BB Menu: **Marker Function** > **Analysis Range (Y)**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:FUNCtion:DOMain:Y {FRANge|BDMarker}

:CALCulate:BB[1-1]:TRACe[1-1]:FUNCtion:DOMain:Y?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:FUNCtion:DOMain:Y FRANge"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:FUNCtion:DOMain:Y ?"
30 ENTER 717;A$
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak

Description

This command gets trace data statistics, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*mean*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No

Error	-

Parameter	*std_dev*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*peak_to_peak*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-

Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Multiple
Var= FRANge
Var = SCPI.CALCulate.BB1.TRACe1.FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.DOMain.X

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.DOMain.Y

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.TYPE

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:FUNCtion:STATistics:DATA?

Query Response

{ - - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:FUNCtion:STATistics:DATA ?"
30 ENTER 717;A$
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak

Description

This command gets memory data statistics, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*mean*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No

Error	-

Parameter	*std_dev*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*peak_to_peak*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-

Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Multiple
Var= FRANge
Var = SCPI.CALCulate.BB1.TRACe1.FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.DOMain.X

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.DOMain.Y

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.STATistics.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.TYPE

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:FUNCtion:STATistics:MEMory?

Query Response

{ - - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:FUNCtion:STATistics:MEMory ?"
30 ENTER 717;A$
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.TYPE

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.TYPE = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.TYPE

Description

This command sets/gets Analysis type, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Analysis type
Data Type	Character string type (String)
Range	OFF|STATistics
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.BB1.TRACe1.FUNCtion.TYPE = Var
Var = SCPI.CALCulate.BB1.TRACe1.FUNCtion.TYPE
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.DOMain.X

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.DOMain.Y

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.STATistics.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory

Equivalent Key

BB Menu: **Marker Function** > **Analysis Type**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:FUNCtion:TYPE {OFF|STATistics}

:CALCulate:BB[1-1]:TRACe[1-1]:FUNCtion:TYPE?

Query Response

{OFF|STATistics} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:FUNCtion:TYPE OFF"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:FUNCtion:TYPE ?"
30 ENTER 717;A$
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).HOLD

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).HOLD = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).HOLD

Description

This command sets/gets Data hold, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Data hold
Data Type	Character string type (String)
Range	OFF|MAXimum|MINimum
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.BB1.TRACe1.HOLD = Var
Var = SCPI.CALCulate.BB1.TRACe1.HOLD
```

Equivalent Key

BB Menu: **Trace View** > **Data Hold**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:HOLD {OFF|MAXimum|MINimum}

:CALCulate:BB[1-1]:TRACe[1-1]:HOLD?

Query Response

{OFF|MAXimum|MINimum} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:HOLD OFF"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:HOLD ?"
30 ENTER 717;A$
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.FAIL

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.FAIL

Description

This command gets limit test pass/fail, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test result
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Boolean
Var= 1
Var = SCPI.CALCulate.BB1.TRACe1.LIMit.FAIL
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.FAIL

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.REPort.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.STATe

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:LIMit:FAIL?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:LIMit:FAIL ?"
20 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.LDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.LDATa = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.LDATa

Description

This command sets/gets expanded lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Lower limit line
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.BB1.TRACe1.LIMit.LOWer.LDATa = Var
Var = SCPI.CALCulate.BB1.TRACe1.LIMit.LOWer.LDATa
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.FAIL

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.REPort.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.STATe

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:LIMit:LOWer:LDATa {1...1601}

:CALCulate:BB[1-1]:TRACe[1-1]:LIMit:LOWer:LDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:BB1:TRACe1:LIMit:LOWer:LDATa 1,1601"
30 OUTPUT 717;":CALCulate:BB1:TRACe1:LIMit:LOWer:LDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

Description

This command clears lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.BB1.TRACe1.LIMit.LOWer.SEGMent.CLEar

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.FAIL

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.REPort.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.STATe

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:CLEar

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:LIMit:LOWer:SEGMent:CLEar"
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

Description

This command sets/gets number of lower limit segments, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Lower limit segments
Data Type	Long integer type (Long)
Range	1 ~ 100
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.BB1.TRACe1.LIMit.LOWer.SEGMent.COUNt = Var
Var = SCPI.CALCulate.BB1.TRACe1.LIMit.LOWer.SEGMent.COUNt
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.FAIL

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.REPort.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.STATe

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:COUNt {1 ~ 100}

:CALCulate:BB[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:COUNt?

Query Response

{1 ~ 100} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:LIMit:LOWer:SEGMent:COUNt 1 "
20 OUTPUT 717;":CALCulate:BB1:TRACe1:LIMit:LOWer:SEGMent:COUNt ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Description

This command sets/gets lower limit line data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Lower limit data
Data Type	Variant type Array (Range)
Range	1...400
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 400) as Variant
SCPI.CALCulate.BB1.TRACe1.LIMit.LOWer.SEGMent.DATA = Var
Var = SCPI.CALCulate.BB1.TRACe1.LIMit.LOWer.SEGMent.DATA
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.FAIL

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.REPort.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.STATe

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:DATA {1...400}

:CALCulate:BB[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:DATA?

Query Response

{1...400} <newline><^END>

Example of use

```
10 Dim A(1:400)
20 OUTPUT 717;":CALCulate:BB1:TRACe1:LIMit:LOWer:SEGMent:DATA 1,400"
30 OUTPUT 717;":CALCulate:BB1:TRACe1:LIMit:LOWer:SEGMent:DATA ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.REPort.DATA

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.REPort.DATA

Description

This command gets pass/fail list, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test report
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
Var = SCPI.CALCulate.BB1.TRACe1.LIMit.REPort.DATA
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.FAIL

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.STATe

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:LIMit:REPort[:DATA]?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:BB1:TRACe1:LIMit:REPort[:DATA] ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.STATe = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.STATe

Description

This command sets/gets limit test on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.BB1.TRACe1.LIMit.STATe = Var
Var = SCPI.CALCulate.BB1.TRACe1.LIMit.STATe
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.FAIL

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.REPort.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

BB Menu: **Display** > **Limit Test** > **Limit Test**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:LIMit[:STATe] {ON|OFF|1|0}

:CALCulate:BB[1-1]:TRACe[1-1]:LIMit[:STATe]?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:LIMit[:STATe] 1"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:LIMit[:STATe] ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.LDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.LDATa = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.LDATa

Description

This command sets/gets expanded upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Upper limit line
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.BB1.TRACe1.LIMit.UPPer.LDATa = Var
Var = SCPI.CALCulate.BB1.TRACe1.LIMit.UPPer.LDATa
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.FAIL

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.REPort.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.STATe

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:LIMit:UPPer:LDATa {1...1601}

:CALCulate:BB[1-1]:TRACe[1-1]:LIMit:UPPer:LDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:BB1:TRACe1:LIMit:UPPer:LDATa 1,1601"
30 OUTPUT 717;":CALCulate:BB1:TRACe1:LIMit:UPPer:LDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

Description

This command clears upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.BB1.TRACe1.LIMit.UPPer.SEGMent.CLEar

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.FAIL

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.REPort.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.STATe

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:CLEar

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:LIMit:UPPer:SEGMent:CLEar"
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Description

This command sets/gets number of upper limit segments, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Upper limit segment
Data Type	Long integer type (Long)
Range	1 ~ 100
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.BB1.TRACe1.LIMit.UPPer.SEGMent.COUNt = Var
Var = SCPI.CALCulate.BB1.TRACe1.LIMit.UPPer.SEGMent.COUNt
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.FAIL

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.REPort.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.STATe

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:COUNt {1 ~ 100}

:CALCulate:BB[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:COUNt?

Query Response

{1 ~ 100} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:LIMit:UPPer:SEGMent:COUNt 1 "
20 OUTPUT 717;":CALCulate:BB1:TRACe1:LIMit:UPPer:SEGMent:COUNt ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Description

This command sets/gets upper limit line data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Upper limit line data
Data Type	Variant type Array (Range)
Range	1...400
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 400) as Variant
SCPI.CALCulate.BB1.TRACe1.LIMit.UPPer.SEGMent.DATA = Var
Var = SCPI.CALCulate.BB1.TRACe1.LIMit.UPPer.SEGMent.DATA
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.FAIL

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.REPort.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.STATe

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:DATA {1...400}

:CALCulate:BB[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:DATA?

Query Response

{1...400} <newline><^END>

Example of use

```
10 Dim A(1:400)
20 OUTPUT 717;":CALCulate:BB1:TRACe1:LIMit:UPPer:SEGMent:DATA 1,400"
30 OUTPUT 717;":CALCulate:BB1:TRACe1:LIMit:UPPer:SEGMent:DATA ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

Description

This command sets marker search LPEak, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.BB1.TRACe1.MARKer10.SEARch.EXECute.LPEak

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

BB Menu: **Marker Search** > **Peak** > **Search Left**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:LPEak

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:SEARch:EXECute:LPEak"
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

Description

This command sets marker search LSPurious, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

This function is allowed to use when the spurious display is power.

Examples

SCPI.CALCulate.BB1.TRACe1.MARKer10.SEARch.EXECute.LSPurious

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

BB Menu: **Marker Search** > **Spurious** > **Search Left**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute.LSPurious

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:SEARch:EXECute.LSPurious"
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

Description

This command sets marker search LTARget, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.BB1.TRACe1.MARKer10.SEARch.EXECute.LTARget

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

BB Menu: **Marker Search** > **Target** > **Search Left**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:LTARget

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:SEARch:EXECute:LTARget"
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

Description

This command sets marker search MAXimum, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.BB1.TRACe1.MARKer10.SEARch.EXECute.MAXimum

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

BB Menu: **Marker Search** > **Search Max**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:MAXimum

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:SEARch:EXECute:MAXimum"
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

Description

This command sets marker search MINimum, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.BB1.TRACe1.MARKer10.SEARch.EXECute.MINimum

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

BB Menu: **Marker Search** > **Search Min**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:MINimum

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:SEARch:EXECute:MINimum"
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

Description

This command sets marker search PEAK, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.BB1.TRACe1.MARKer10.SEARch.EXECute.PEAK

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

BB Menu: **Marker Search** > **Peak** > **Search Peak**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:PEAK

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:SEARch:EXECute:PEAK"
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

Description

This command sets marker search RPEak, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.BB1.TRACe1.MARKer10.SEARch.EXECute.RPEak

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

BB Menu: **Marker Search** > **Peak** > **Search Right**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:RPEak

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:SEARch:EXECute:RPEak"
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

Description

This command sets marker search RSPurious, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

This function is allowed to use when the spurious display is power.

Examples

SCPI.CALCulate.BB1.TRACe1.MARKer10.SEARch.EXECute.RSPurious

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

BB Menu: **Marker Search** > **Spurious** > **Search Right**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:RSPurious

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:SEARch:EXECute:RSPurious"
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

Description

This command sets marker search RTARget, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.BB1.TRACe1.MARKer10.SEARch.EXECute.RTARget

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

BB Menu: **Marker Search** > **Target** > **Search Right**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:RTARget

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:SEARch:EXECute:RTARget"
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

Description

This command sets marker search SPURious, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

NOTE This function is allowed to use when the spurious display is power.

Examples

SCPI.CALCulate.BB1.TRACe1.MARKer10.SEARch.EXECute.SPURious

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

BB Menu: **Marker Search** > **Spurious** > **Search Spurious**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:SPURious

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:SEARch:EXECute:SPURious"
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Description

This command sets marker search TARGet, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.BB1.TRACe1.MARKer10.SEARch.EXECute.TARGet

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

Equivalent Key

BB Menu: **Marker Search** > **Target** > **Search Target**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:TARGet

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:SEARch:EXECute:TARGet"
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion

Description

This command sets/gets marker-search-peak excursion value, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search peak excursion value
Data Type	Double precision floating point type (Double)
Range	0 ~ 10G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.BB1.TRACe1.MARKer10.SEARch.PEAK.EXCursion = Var
Var = SCPI.CALCulate.BB1.TRACe1.MARKer10.SEARch.PEAK.EXCursion
```

Related Objects

Equivalent Key

BB Menu: **Marker Search** > **Peak** > **Peak Excursion**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:PEAK:EXCursion {0 ~ 10G}

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:PEAK:EXCursion?

Query Response

{0 ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:SEARch:PEAK:EXCursion 0"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:SEARch:PEAK:EXCursion ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity

Description

This command sets/gets marker-search-peak polarity type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search peak polarity type
Data Type	Character string type (String)
Range	POSitive|NEGative|BOTH
Preset Value	POSitive
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "POSitive"
SCPI.CALCulate.BB1.TRACe1.MARKer10.SEARch.PEAK.POLarity = Var
Var = SCPI.CALCulate.BB1.TRACe1.MARKer10.SEARch.PEAK.POLarity
```

Related Objects

Equivalent Key

BB Menu: **Marker Search** > **Peak** > **Peak Polarity**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:PEAK:POLarity {POSitive|NEGative|BOTH}

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:PEAK:POLarity?

Query Response

{POSitive|NEGative|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:SEARch:PEAK:POLarity POSitive"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:SEARch:PEAK:POLarity ?"
30 ENTER 717;A$
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition

Description

This command sets/gets marker-tar transition type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker target transition type
Data Type	Character string type (String)
Range	POSitive|NEGative|BOTH
Preset Value	BOTH
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as String
Var= "POSitive"
SCPI.CALCulate.BB1.TRACe1.MARKer10.SEARch.TARGet.TRANsition = Var
Var = SCPI.CALCulate.BB1.TRACe1.MARKer10.SEARch.TARGet.TRANsition
```

Related Objects

Equivalent Key

BB Menu: **Marker Search** > **Target** > **Target Transition**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TARGet:TRANsition {POSitive|NEGative|BOTH}

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TARGet:TRANsition?

Query Response

{POSitive|NEGative|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:SEARch:TARGet:TRANsition POSitive"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:SEARch:TARGet:TRANsition ?"
30 ENTER 717;A$
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y

Description

This command sets/gets marker target value, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker target value
Data Type	Double precision floating point type (Double)
Range	-10G ~ 10G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.BB1.TRACe1.MARKer10.SEARch.TARGet.Y = Var
Var = SCPI.CALCulate.BB1.TRACe1.MARKer10.SEARch.TARGet.Y
```

Related Objects

Equivalent Key

BB Menu: **Marker Search** > **Target** > **Target Value**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TARGet:Y {-10G ~ 10G}

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TARGet:Y?

Query Response

{-10G ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:SEARch:TARGet:Y -10000000000"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:SEARch:TARGet:Y ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE

Description

This command sets/gets search tracking type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search tracking type
Data Type	Character string type (String)
Range	OFF|MAXimum|MINimum|PEAK|TARGet|SPURious
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.BB1.TRACe1.MARKer10.SEARch.TRACking.TYPE = Var
Var = SCPI.CALCulate.BB1.TRACe1.MARKer10.SEARch.TRACking.TYPE
```

Related Objects

Equivalent Key

BB Menu: **Marker Search** > **Tracking**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TRACking:TYPE {OFF|MAXimum|MINimum|PEAK|TARGet|SPURious}

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TRACking:TYPE?

Query Response

{OFF|MAXimum|MINimum|PEAK|TARGet|SPURious} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:SEARch:TRACking:TYPE OFF"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:SEARch:TRACking:TYPE ?"
30 ENTER 717;A$
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).STATe = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).STATe

Description

This command sets/gets marker visible on/off, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker visible state
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.BB1.TRACe1.MARKer10.STATe = Var
Var = SCPI.CALCulate.BB1.TRACe1.MARKer10.STATe
```

Equivalent Key

BB Menu: **Marker** > **Clear Marker Menu** > **Marker 1**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:STATe {ON|OFF|1|0}

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:STATe 1"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).X = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).X

Description

This command sets/gets marker x position, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker X position
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	1k
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
SCPI.CALCulate.BB1.TRACe1.MARKer10.X = Var
Var = SCPI.CALCulate.BB1.TRACe1.MARKer10.X
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).Y

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:X

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:X?

Query Response

{Double} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:X 0"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:X ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).Y

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).Y

Description

This command gets marker y position, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker Y position
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var = SCPI.CALCulate.BB1.TRACe1.MARKer10.Y
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).MARKer(Mk).X

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:MARKer[1-10]:Y?

Query Response

{Double} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:MARKer10:Y ?"
20 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).MATH.FUNCtion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).MATH.FUNCtion = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).MATH.FUNCtion

Description

This command sets/gets math operation type, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Math function operation
Data Type	Character string type (String)
Range	NORMal|SUBTract|DIVide|ADD|MULTiply|USER
Preset Value	NORMal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NORMal"
SCPI.CALCulate.BB1.TRACe1.MATH.FUNCtion = Var
Var = SCPI.CALCulate.BB1.TRACe1.MATH.FUNCtion
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).MATH.MEMorize

SCPI.CALCulate.BB(Ch).TRACe(Tr).MATH.OFFSet

Equivalent Key

BB Menu: **Trace View** > **Data Math**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:MATH:FUNCtion {NORMal|SUBTract|DIVide|ADD|MULTiply|USER}

:CALCulate:BB[1-1]:TRACe[1-1]:MATH:FUNCtion?

Query Response

{NORMal|SUBTract|DIVide|ADD|MULTiply|USER} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:MATH:FUNCtion NORMal"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:MATH:FUNCtion ?"
30 ENTER 717;A$
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).MATH.MEMorize

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).MATH.MEMorize

Description

This command copies data to memory, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.BB1.TRACe1.MATH.MEMorize

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).MATH.FUNCtion

SCPI.CALCulate.BB(Ch).TRACe(Tr).MATH.OFFSet

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:MATH:MEMorize

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:MATH:MEMorize"
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).MATH.OFFSet

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).MATH.OFFSet = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).MATH.OFFSet

Description

This command sets/gets Trace Offset, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace offset
Data Type	Double precision floating point type (Double)
Range	-500G ~ 500G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -500000000000
SCPI.CALCulate.BB1.TRACe1.MATH.OFFSet = Var
Var = SCPI.CALCulate.BB1.TRACe1.MATH.OFFSet
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).MATH.FUNCtion

SCPI.CALCulate.BB(Ch).TRACe(Tr).MATH.MEMorize

Equivalent Key

BB Menu: **Trace View** > **Offset**

BB Menu: **Trace View** > **Marker** > **-Offset**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:MATH:OFFSet {-500G ~ 500G}

:CALCulate:BB[1-1]:TRACe[1-1]:MATH:OFFSet?

Query Response

{-500G ~ 500G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:MATH:OFFSet -500000000000"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:MATH:OFFSet ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).SMOothing.APERture

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).SMOothing.APERture = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).SMOothing.APERture

Description

This command sets/gets Smoothing aperture, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Smoothing aperture
Data Type	Double precision floating point type (Double)
Range	50m ~ 25
Preset Value	1.5
Unit	%
Resolution	10m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0.05
SCPI.CALCulate.BB1.TRACe1.SMOothing.APERture = Var
Var = SCPI.CALCulate.BB1.TRACe1.SMOothing.APERture
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).SMOothing.STATe

Equivalent Key

BB Menu: **Trace View** > **Aperture**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:SMOothing:APERture {50m ~ 25}

:CALCulate:BB[1-1]:TRACe[1-1]:SMOothing:APERture?

Query Response

{50m ~ 25} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:SMOothing:APERture 0.05"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:SMOothing:APERture ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).SMOothing.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).SMOothing.STATe = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).SMOothing.STATe

Description

This command sets/gets Smoothing on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Smoothing (On/OFF)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.BB1.TRACe1.SMOothing.STATe = Var
Var = SCPI.CALCulate.BB1.TRACe1.SMOothing.STATe
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).SMOothing.APERture

Equivalent Key

BB Menu: **Trace View** > **Smoothing**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:SMOothing:STATe {ON|OFF|1|0}

:CALCulate:BB[1-1]:TRACe[1-1]:SMOothing:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:SMOothing:STATe 1"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:SMOothing:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OMISsion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OMISsion = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OMISsion

Description

This command sets/gets Spurious display omission ON/OFF, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious display omission
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.BB1.TRACe1.SPURious.OMISsion = Var
Var = SCPI.CALCulate.BB1.TRACe1.SPURious.OMISsion
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:SPURious:OMISsion {ON|OFF|1|0}

:CALCulate:BB[1-1]:TRACe[1-1]:SPURious:OMISsion?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:SPURious:OMISsion 1"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:SPURious:OMISsion ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

Description

This command clears threshold table, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.BB1.TRACe1.SPURious.OSSPur.FLISt.CLEar

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:SPURious:OSSPur:FLISt:CLEar

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:SPURious:OSSPur:FLISt:CLEar"
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

Description

This command sets/gets Spurious Table for Omit Specified Spurious, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious table for omit specified spurious
Data Type	Long integer type (Long)
Range	1 ~ 100
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.BB1.TRACe1.SPURious.OSSPur.FLISt.COUNt = Var
Var = SCPI.CALCulate.BB1.TRACe1.SPURious.OSSPur.FLISt.COUNt
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:SPURious:OSSPur:FLISt:COUNt {1 ~ 100}

:CALCulate:BB[1-1]:TRACe[1-1]:SPURious:OSSPur:FLISt:COUNt?

Query Response

{1 ~ 100} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:SPURious:OSSPur:FLISt:COUNt 1 "
20 OUTPUT 717;":CALCulate:BB1:TRACe1:SPURious:OSSPur:FLISt:COUNt ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

Description

This command sets/gets threshold table, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Threshold table
Data Type	Variant type Array (Range)
Range	1...100
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 100) as Variant
SCPI.CALCulate.BB1.TRACe1.SPURious.OSSPur.FLISt.DATA = Var
Var = SCPI.CALCulate.BB1.TRACe1.SPURious.OSSPur.FLISt.DATA
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:SPURious:OSSPur:FLISt:DATA {1...100}

:CALCulate:BB[1-1]:TRACe[1-1]:SPURious:OSSPur:FLISt:DATA?

Query Response

{1...100} <newline><^END>

Example of use

```
10 Dim A(1:100)
20 OUTPUT 717;":CALCulate:BB1:TRACe1:SPURious:OSSPur:FLISt:DATA 1,100"
30 OUTPUT 717;":CALCulate:BB1:TRACe1:SPURious:OSSPur:FLISt:DATA ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

Description

This command sets/gets Specified Spurious display omission ON/OFF, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious display mission for the selected trace
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.BB1.TRACe1.SPURious.OSSPur.OMISsion = Var
Var = SCPI.CALCulate.BB1.TRACe1.SPURious.OSSPur.OMISsion
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:SPURious:OSSPur:OMISsion {ON|OFF|1|0}

:CALCulate:BB[1-1]:TRACe[1-1]:SPURious:OSSPur:OMISsion?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:SPURious:OSSPur:OMISsion 1"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:SPURious:OSSPur:OMISsion ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.POWer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.POWer = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.POWer

Description

This command sets/gets Spurious display merging ON/OFF, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious display merging
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.BB1.TRACe1.SPURious.POWer = Var
Var = SCPI.CALCulate.BB1.TRACe1.SPURious.POWer
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:SPURious:POWer {ON|OFF|1|0}

:CALCulate:BB[1-1]:TRACe[1-1]:SPURious:POWer?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:SPURious:POWer 1"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:SPURious:POWer ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.SENSibility

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.SENSibility = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.SENSibility

Description

This command sets/gets spurious sensibility, for the selected trace *Tr* of the selected channel *Ch*.

The low peak has real sensibility when the value is small.

Variable

Parameter	*Value*
Description	Spurious sensibility
Data Type	Double precision floating point type (Double)
Range	10m ~ 10
Preset Value	3
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 5
SCPI.CALCulate.BB1.TRACe1.SPURious.SENSibility = Var
Var = SCPI.CALCulate.BB1.TRACe1.SPURious.SENSibility
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).SMOothing.APERture

SCPI.CALCulate.BB(Ch).TRACe(Tr).SMOothing.STATe

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

BB Menu: **Trace View** > **Spurious** > **Spur Sensibility**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:SPURious:SENSibility {10m ~ 10}

:CALCulate:BB[1-1]:TRACe[1-1]:SPURious:SENSibility?

Query Response

{10m ~ 10} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:SPURious:SENSibility 5"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:SPURious:SENSibility?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.SLISt

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.SLISt

Description

This command gets Spurious List, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious List
Data Type	Variant type Array (Range)
Range	1...0
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 0) as Variant
Var = SCPI.CALCulate.BB1.TRACe1.SPURious.SLISt
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:SPURious:SLISt?

Query Response

{1...0} <newline><^END>

Example of use

```
10 Dim A(1:0)
20 OUTPUT 717;":CALCulate:BB1:TRACe1:SPURious:SLISt ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

Description

This command sets/gets Spurious minimum level definition, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious minimum level definition
Data Type	Double precision floating point type (Double)
Range	-500 ~ 500
Preset Value	-500
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.BB1.TRACe1.SPURious.THReshold.LEVel.MINimum = Var
Var = SCPI.CALCulate.BB1.TRACe1.SPURious.THReshold.LEVel.MINimum
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

BB Menu: **Trace View** > **Spurious** > **Minimum Spur Level**

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:SPURious:THReshold:LEVel:MINimum {-500 ~ 500}

:CALCulate:BB[1-1]:TRACe[1-1]:SPURious:THReshold:LEVel:MINimum?

Query Response

{-500 ~ 500} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:SPURious:THReshold:LEVel:MINimum 0"
20 OUTPUT 717;":CALCulate:BB1:TRACe1:SPURious:THReshold:LEVel:MINimum ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

Description

This command clears threshold table, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.BB1.TRACe1.SPURious.THReshold.TABLe.CLEar

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:CLEar

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:SPURious:THReshold:TABLe:CLEar"
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

Description

This command sets/gets number of threshold segments, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Number of threshold segments
Data Type	Long integer type (Long)
Range	1 ~ 20
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.BB1.TRACe1.SPURious.THReshold.TABLe.COUNt = Var
Var = SCPI.CALCulate.BB1.TRACe1.SPURious.THReshold.TABLe.COUNt
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:COUNt {1 ~ 20}

:CALCulate:BB[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:COUNt?

Query Response

{1 ~ 20} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:BB1:TRACe1:SPURious:THReshold:TABLe:COUNt 1 "
20 OUTPUT 717;":CALCulate:BB1:TRACe1:SPURious:THReshold:TABLe:COUNt ?"
30 ENTER 717;A
```

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA = *Value*

Value = SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Description

This command sets/gets threshold table, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Threshold table
Data Type	Variant type Array (Range)
Range	1...60
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 60) as Variant
SCPI.CALCulate.BB1.TRACe1.SPURious.THReshold.TABLe.DATA = Var
Var = SCPI.CALCulate.BB1.TRACe1.SPURious.THReshold.TABLe.DATA
```

Related Objects

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.BB(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:BB[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:DATA {1...60}

:CALCulate:BB[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:DATA?

Query Response

{1...60} <newline><^END>

Example of use

```
10 Dim A(1:60)
20 OUTPUT 717;":CALCulate:BB1:TRACe1:SPURious:THReshold:TABLe:DATA 1,60"
30 OUTPUT 717;":CALCulate:BB1:TRACe1:SPURious:THReshold:TABLe:DATA ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.FP(Ch).ALLTrace.ACTive

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).ALLTrace.ACTive = *Value*

Value = SCPI.CALCulate.FP(Ch).ALLTrace.ACTive

Description

This command sets/gets active trace, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Active trace
Data Type	Long integer type (Long)
Range	1 ~ 4
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
```

```
SCPI.CALCulate.FP1.ALLTrace.ACTive = Var
Var = SCPI.CALCulate.FP1.ALLTrace.ACTive
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:ALLTrace:ACTive {1 ~ 4}

:CALCulate:FP[1-1]:ALLTrace:ACTive?

Query Response

{1 ~ 4} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:ALLTrace:ACTive 1 "
20 OUTPUT 717;":CALCulate:FP1:ALLTrace:ACTive ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).ALLTrace.BDMarker.X.COUPle.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).ALLTrace.BDMarker.X.COUPle.STATe = *Value*

Value = SCPI.CALCulate.FP(Ch).ALLTrace.BDMarker.X.COUPle.STATe

Description

This command sets/gets x band marker couple on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	X band marker couple (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.FP1.ALLTrace.BDMarker.X.COUPle.STATe = Var
Var = SCPI.CALCulate.FP1.ALLTrace.BDMarker.X.COUPle.STATe
```

Equivalent Key

FP Menu: **Marker Search** > **Couple**

FP Menu: **Marker Function** > **Couple**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:ALLTrace:BDMarker:X:COUPle:STATe {ON|OFF|1|0}

:CALCulate:FP[1-1]:ALLTrace:BDMarker:X:COUPle:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:ALLTrace:BDMarker:X:COUPle:STATe 1"
20 OUTPUT 717;":CALCulate:FP1:ALLTrace:BDMarker:X:COUPle:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).ALLTrace.LIMit.FAIL

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.FP(Ch).ALLTrace.LIMit.FAIL

Description

This command gets limit test pass/fail, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Result of limit test
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Boolean
Var= 1
Var = SCPI.CALCulate.FP1.ALLTrace.LIMit.FAIL
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:ALLTrace:LIMit:FAIL?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:ALLTrace:LIMit:FAIL ?"
20 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.COUPle.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.COUPle.STATe = *Value*

Value = SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.COUPle.STATe

Description

This command sets/gets marker couple on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker Coupling Function state (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.FP1.ALLTrace.MARKer.COUPle.STATe = Var
Var = SCPI.CALCulate.FP1.ALLTrace.MARKer.COUPle.STATe
```

Related Objects

SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.DISCrete.STATe

SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.REFerence.NUMBer

SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.REFerence.STATe

Equivalent Key

FP Menu: **Marker** > **Couple**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:ALLTrace:MARKer:COUPle:STATe {ON|OFF|1|0}

:CALCulate:FP[1-1]:ALLTrace:MARKer:COUPle:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:ALLTrace:MARKer:COUPle:STATe 1"
20 OUTPUT 717;":CALCulate:FP1:ALLTrace:MARKer:COUPle:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.DISCrete.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.DISCrete.STATe = *Value*

Value = SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.DISCrete.STATe

Description

This command sets/gets marker discrete on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Enables/Disables marker discrete
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.FP1.ALLTrace.MARKer.DISCrete.STATe = Var
Var = SCPI.CALCulate.FP1.ALLTrace.MARKer.DISCrete.STATe
```

Related Objects

SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.COUPle.STATe

SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.REFerence.NUMBer

SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.REFerence.STATe

Equivalent Key

FP Menu: **Marker** > **More Functions** > **Discrete**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:ALLTrace:MARKer:DISCrete:STATe {ON|OFF|1|0}

:CALCulate:FP[1-1]:ALLTrace:MARKer:DISCrete:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:ALLTrace:MARKer:DISCrete:STATe 1"
20 OUTPUT 717;":CALCulate:FP1:ALLTrace:MARKer:DISCrete:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.REFerence.NUMBer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.REFerence.NUMBer = *Value*

Value = SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.REFerence.NUMBer

Description

This command sets/gets reference marker number., for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker reference number
Data Type	Long integer type (Long)
Range	1 ~ 10
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.FP1.ALLTrace.MARKer.REFerence.NUMBer = Var
Var = SCPI.CALCulate.FP1.ALLTrace.MARKer.REFerence.NUMBer
```

Related Objects

SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.COUPle.STATe

SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.DISCrete.STATe

SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.REFerence.STATe

Equivalent Key

FP Menu: **Marker** > **More Functions** > **Ref Marker**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:ALLTrace:MARKer:REFerence:NUMBer {1 ~ 10}

:CALCulate:FP[1-1]:ALLTrace:MARKer:REFerence:NUMBer?

Query Response

{1 ~ 10} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:ALLTrace:MARKer:REFerence:NUMBer 1 "
20 OUTPUT 717;":CALCulate:FP1:ALLTrace:MARKer:REFerence:NUMBer ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.REFerence.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.REFerence.STATe = *Value*

Value = SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.REFerence.STATe

Description

This command sets/gets delta marker mode on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Delta marker mode (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.FP1.ALLTrace.MARKer.REFerence.STATe = Var
Var = SCPI.CALCulate.FP1.ALLTrace.MARKer.REFerence.STATe
```

Related Objects

SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.COUPle.STATe

SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.DISCrete.STATe

SCPI.CALCulate.FP(Ch).ALLTrace.MARKer.REFerence.NUMBer

Equivalent Key

FP Menu: **Marker** > **More Functions** > **Ref Marker Mode**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:ALLTrace:MARKer:REFerence:STATe {ON|OFF|1|0}

:CALCulate:FP[1-1]:ALLTrace:MARKer:REFerence:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:ALLTrace:MARKer:REFerence:STATe 1"
20 OUTPUT 717;":CALCulate:FP1:ALLTrace:MARKer:REFerence:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).DATA.RDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).DATA.RDATa = *Value*

Value = SCPI.CALCulate.FP(Ch).DATA.RDATa

Description

This command sets/gets measurement raw data, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Measurement raw power data
Data Type	Variant type Array (Range)
Range	1...3003
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 3003) as Variant
SCPI.CALCulate.FP1.DATA.RDATa = Var
Var = SCPI.CALCulate.FP1.DATA.RDATa
```

Related Objects

SCPI.CALCulate.FP(Ch).DATA.TDATa

SCPI.CALCulate.FP(Ch).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:DATA:RDATa {1...3003}

:CALCulate:FP[1-1]:DATA:RDATa?

Query Response

{1...3003} <newline><^END>

Example of use

```
10 Dim A(1:3003)
20 OUTPUT 717;":CALCulate:FP1:DATA:RDATa 1,3003"
30 OUTPUT 717;":CALCulate:FP1:DATA:RDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.FP(Ch).DATA.TDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).DATA.TDATa = *Value*

Value = SCPI.CALCulate.FP(Ch).DATA.TDATa

Description

This command sets/gets Tester mode data, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Tester mode data; frequency, power and current
Data Type	Variant type Array (Range)
Range	1...3
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 3) as Variant
SCPI.CALCulate.FP1.DATA.TDATa = Var
Var = SCPI.CALCulate.FP1.DATA.TDATa
```

Related Objects

SCPI.CALCulate.FP(Ch).DATA.RDATa

SCPI.CALCulate.FP(Ch).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:DATA:TDATa {1...3}

:CALCulate:FP[1-1]:DATA:TDATa?

Query Response

{1...3} <newline><^END>

Example of use

```
10 Dim A(1:3)
20 OUTPUT 717;":CALCulate:FP1:DATA:TDATa 1,3"
30 OUTPUT 717;":CALCulate:FP1:DATA:TDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.FP(Ch).DATA.XDATa

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.FP(Ch).DATA.XDATa

Description

This command gets X axis data, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	X axis data
Data Type	Variant type Array (Range)
Range	1...1001
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1001) as Variant
Var = SCPI.CALCulate.FP1.DATA.XDATa
```

Related Objects

SCPI.CALCulate.FP(Ch).DATA.RDATa

SCPI.CALCulate.FP(Ch).DATA.TDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:DATA:XDATa?

Query Response

{1...1001} <newline><^END>

Example of use

```
10 Dim A(1:1001)
20 OUTPUT 717;":CALCulate:FP1:DATA:XDATa ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).ALLMarker.ACTive

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).ALLMarker.ACTive = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).ALLMarker.ACTive

Description

This command sets/gets active marker, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Active marker
Data Type	Long integer type (Long)
Range	1 ~ 10
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.FP1.TRACe4.ALLMarker.ACTive = Var
Var = SCPI.CALCulate.FP1.TRACe4.ALLMarker.ACTive
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:ALLMarker:ACTive {1 ~ 10}

:CALCulate:FP[1-1]:TRACe[1-4]:ALLMarker:ACTive?

Query Response

{1 ~ 10} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:ALLMarker:ACTive 1 "
20 OUTPUT 717;":CALCulate:FP1:TRACe4:ALLMarker:ACTive ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

Description

This command sets/gets marker search X range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search X range source
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	FRANge
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.FP1.TRACe4.ALLMarker.SEARch.DOMain.X = Var
Var = SCPI.CALCulate.FP1.TRACe4.ALLMarker.SEARch.DOMain.X
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

SCPI.CALCulate.FP(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Equivalent Key

FP Menu: **Marker Search** > **Search Range (X)**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:ALLMarker:SEARch:DOMain:X {FRANge|BDMarker}

:CALCulate:FP[1-1]:TRACe[1-4]:ALLMarker:SEARch:DOMain:X?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:ALLMarker:SEARch:DOMain:X FRANge"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:ALLMarker:SEARch:DOMain:X ?"
30 ENTER 717;A$
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Description

This command sets/gets marker search Y range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search Y range
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	FRANge
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.FP1.TRACe4.ALLMarker.SEARch.DOMain.Y = Var
Var = SCPI.CALCulate.FP1.TRACe4.ALLMarker.SEARch.DOMain.Y
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

SCPI.CALCulate.FP(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Equivalent Key

FP Menu: **Marker Search** > **Search Range (Y)**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:ALLMarker:SEARch:DOMain:Y {FRANge|BDMarker}

:CALCulate:FP[1-1]:TRACe[1-4]:ALLMarker:SEARch:DOMain:Y?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:ALLMarker:SEARch:DOMain:Y FRANge"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:ALLMarker:SEARch:DOMain:Y ?"
30 ENTER 717;A$
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Description

This command sets search peak all, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.FP1.TRACe4.ALLMarker.SEARch.PEAK

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

SCPI.CALCulate.FP(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Equivalent Key

FP Menu: **Marker Search** > **Peak** > **Search Peak All**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:ALLMarker:SEARch:PEAK

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:ALLMarker:SEARch:PEAK"
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.CENTer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.CENTer = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.CENTer

Description

This command sets/gets band marker X center, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X center
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.FP1.TRACe4.BDMarker.X.CENTer = Var
Var = SCPI.CALCulate.FP1.TRACe4.BDMarker.X.CENTer
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

FP Menu: **Marker Search** > **Band Marker X** > **Center**

FP Menu: **Marker Function** > **Band Marker X** > **Center**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:BDMarker:X:CENTer {-1T ~ 1T}

:CALCulate:FP[1-1]:TRACe[1-4]:BDMarker:X:CENTer?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:BDMarker:X:CENTer -10000000000"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:BDMarker:X:CENTer ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.SPAN

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.SPAN = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.SPAN

Description

This command sets/gets band marker X span, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X span
Data Type	Double precision floating point type (Double)
Range	0 ~ 2T
Preset Value	2T
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.FP1.TRACe4.BDMarker.X.SPAN = Var
Var = SCPI.CALCulate.FP1.TRACe4.BDMarker.X.SPAN
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

FP Menu: **Marker Search** > **Band Marker X** > **Span**

FP Menu: **Marker Function** > **Band Marker X** > **Span**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:BDMarker:X:SPAN {0 ~ 2T}

:CALCulate:FP[1-1]:TRACe[1-4]:BDMarker:X:SPAN?

Query Response

{0 ~ 2T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:BDMarker:X:SPAN 0"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:BDMarker:X:SPAN ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.STARt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.STARt = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.STARt

Description

This command sets/gets band marker X start, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X start
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	-1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.FP1.TRACe4.BDMarker.X.STARt = Var
Var = SCPI.CALCulate.FP1.TRACe4.BDMarker.X.STARt
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

FP Menu: **Marker Search** > **Band Marker X** > **Start**

FP Menu: **Marker Function** > **Band Marker X** > **Start**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:BDMarker:X:STARt {-1T ~ 1T}

:CALCulate:FP[1-1]:TRACe[1-4]:BDMarker:X:STARt?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:BDMarker:X:STARt -10000000000"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:BDMarker:X:STARt ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.STATe = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.STATe

Description

This command sets/gets band marker visible on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker visibility (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.FP1.TRACe4.BDMarker.X.STATe = Var
Var = SCPI.CALCulate.FP1.TRACe4.BDMarker.X.STATe
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

FP Menu: **Marker Search** > **Band Marker X** > **Band Marker X**

FP Menu: **Marker Function** > **Band Marker X** > **Band Marker X**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:BDMarker:X:STATe {ON|OFF|1|0}

:CALCulate:FP[1-1]:TRACe[1-4]:BDMarker:X:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:BDMarker:X:STATe 1"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:BDMarker:X:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.STOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.STOP = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.STOP

Description

This command sets/gets band marker X stop, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X stop
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.FP1.TRACe4.BDMarker.X.STOP = Var
Var = SCPI.CALCulate.FP1.TRACe4.BDMarker.X.STOP
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.X.STATe

Equivalent Key

FP Menu: **Marker Search** > **Band Marker X** > **Stop**

FP Menu: **Marker Function** > **Band Marker X** > **Stop**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:BDMarker:X:STOP {-1T ~ 1T}

:CALCulate:FP[1-1]:TRACe[1-4]:BDMarker:X:STOP?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:BDMarker:X:STOP -10000000000"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:BDMarker:X:STOP ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.CENTer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.CENTer = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.CENTer

Description

This command sets/gets band marker Y center, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y center
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.FP1.TRACe4.BDMarker.Y.CENTer = Var
Var = SCPI.CALCulate.FP1.TRACe4.BDMarker.Y.CENTer
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

FP Menu: **Marker Search** > **Band Marker Y** > **Center**

FP Menu: **Marker Function** > **Band Marker Y** > **Center**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:BDMarker:Y:CENTer {-1T ~ 1T}

:CALCulate:FP[1-1]:TRACe[1-4]:BDMarker:Y:CENTer?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:BDMarker:Y:CENTer -10000000000"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:BDMarker:Y:CENTer ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.SPAN

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.SPAN = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.SPAN

Description

This command sets/gets band marker Y span, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y span
Data Type	Double precision floating point type (Double)
Range	0 ~ 2T
Preset Value	2T
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.FP1.TRACe4.BDMarker.Y.SPAN = Var
Var = SCPI.CALCulate.FP1.TRACe4.BDMarker.Y.SPAN
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

FP Menu: **Marker Search** > **Band Marker Y** > **Span**

FP Menu: **Marker Function** > **Band Marker Y** > **Span**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:BDMarker:Y:SPAN {0 ~ 2T}

:CALCulate:FP[1-1]:TRACe[1-4]:BDMarker:Y:SPAN?

Query Response

{0 ~ 2T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:BDMarker:Y:SPAN 0"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:BDMarker:Y:SPAN ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.STARt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.STARt = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.STARt

Description

This command sets/gets band marker Y start, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y start
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	-1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.FP1.TRACe4.BDMarker.Y.STARt = Var
Var = SCPI.CALCulate.FP1.TRACe4.BDMarker.Y.STARt
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

FP Menu: **Marker Search** > **Band Marker Y** > **Start**

FP Menu: **Marker Function** > **Band Marker Y** > **Start**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:BDMarker:Y:STARt {-1T ~ 1T}

:CALCulate:FP[1-1]:TRACe[1-4]:BDMarker:Y:STARt?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:BDMarker:Y:STARt -10000000000"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:BDMarker:Y:STARt ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.STATe = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.STATe

Description

This command sets/gets band marker visible on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y state
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.FP1.TRACe4.BDMarker.Y.STATe = Var
Var = SCPI.CALCulate.FP1.TRACe4.BDMarker.Y.STATe
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

FP Menu: **Marker Search** > **Band Marker Y** > **Band Marker Y**

FP Menu: **Marker Function** > **Band Marker Y** > **Band Marker Y**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:BDMarker:Y:STATe {ON|OFF|1|0}

:CALCulate:FP[1-1]:TRACe[1-4]:BDMarker:Y:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:BDMarker:Y:STATe 1"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:BDMarker:Y:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.STOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.STOP = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.STOP

Description

This command sets/gets band marker Y stop, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y stop
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.FP1.TRACe4.BDMarker.Y.STOP = Var
Var = SCPI.CALCulate.FP1.TRACe4.BDMarker.Y.STOP
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.FP(Ch).TRACe(Tr).BDMarker.Y.STATe

Equivalent Key

FP Menu: **Marker Search** > **Band Marker Y** > **Stop**

FP Menu: **Marker Function** > **Band Marker Y** > **Stop**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:BDMarker:Y:STOP {-1T ~ 1T}

:CALCulate:FP[1-1]:TRACe[1-4]:BDMarker:Y:STOP?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:BDMarker:Y:STOP -10000000000"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:BDMarker:Y:STOP ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.COPY

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.COPY = *Value*

Description

This command copies data to user trace, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Copies data to user trace
Data Type	Long integer type (Long)
Range	1 ~ 8
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.FP1.TRACe4.DATA.COPY = Var
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:DATA:COPY {1 ~ 8}

Query Response

{1 ~ 8} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:DATA:COPY 1 "
20 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.FDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.FDATa = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.FDATa

Description

This command sets/gets formatted trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Formatted trace data
Data Type	Variant type Array (Range)
Range	1...1001
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1001) as Variant
SCPI.CALCulate.FP1.TRACe4.DATA.FDATa = Var
Var = SCPI.CALCulate.FP1.TRACe4.DATA.FDATa
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:DATA:FDATa {1...1001}

:CALCulate:FP[1-1]:TRACe[1-4]:DATA:FDATa?

Query Response

{1...1001} <newline><^END>

Example of use

```
10 Dim A(1:1001)
20 OUTPUT 717;":CALCulate:FP1:TRACe4:DATA:FDATa 1,1001"
30 OUTPUT 717;":CALCulate:FP1:TRACe4:DATA:FDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.FMEMory

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.FMEMory = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.FMEMory

Description

This command sets/gets formatted memory data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Formatted memory data
Data Type	Variant type Array (Range)
Range	1...1001
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1001) as Variant
SCPI.CALCulate.FP1.TRACe4.DATA.FMEMory = Var
Var = SCPI.CALCulate.FP1.TRACe4.DATA.FMEMory
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:DATA:FMEMory {1...1001}

:CALCulate:FP[1-1]:TRACe[1-4]:DATA:FMEMory?

Query Response

{1...1001} <newline><^END>

Example of use

```
10 Dim A(1:1001)
20 OUTPUT 717;":CALCulate:FP1:TRACe4:DATA:FMEMory 1,1001"
30 OUTPUT 717;":CALCulate:FP1:TRACe4:DATA:FMEMory ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.UDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.UDATa = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.UDATa

Description

This command sets/gets unformatted trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Unformatted trace data
Data Type	Variant type Array (Range)
Range	1...1001
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1001) as Variant
SCPI.CALCulate.FP1.TRACe4.DATA.UDATa = Var
Var = SCPI.CALCulate.FP1.TRACe4.DATA.UDATa
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:DATA:UDATa {1...1001}

:CALCulate:FP[1-1]:TRACe[1-4]:DATA:UDATa?

Query Response

{1...1001} <newline><^END>

Example of use

```
10 Dim A(1:1001)
20 OUTPUT 717;":CALCulate:FP1:TRACe4:DATA:UDATa 1,1001"
30 OUTPUT 717;":CALCulate:FP1:TRACe4:DATA:UDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.UMEMory

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.UMEMory = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.UMEMory

Description

This command sets/gets unformatted memory data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Unformatted memory data
Data Type	Variant type Array (Range)
Range	1...1001
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1001) as Variant
SCPI.CALCulate.FP1.TRACe4.DATA.UMEMory = Var
Var = SCPI.CALCulate.FP1.TRACe4.DATA.UMEMory
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.FP(Ch).TRACe(Tr).DATA.UDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:DATA:UMEMory {1...1001}

:CALCulate:FP[1-1]:TRACe[1-4]:DATA:UMEMory?

Query Response

{1...1001} <newline><^END>

Example of use

```
10 Dim A(1:1001)
20 OUTPUT 717;":CALCulate:FP1:TRACe4:DATA:UMEMory 1,1001"
30 OUTPUT 717;":CALCulate:FP1:TRACe4:DATA:UMEMory ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).EQUation.TEXT

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).EQUation.TEXT = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).EQUation.TEXT

Description

This command sets/gets equation string, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Equation string
Data Type	Character string type (String)
Range	254 chars
Preset Value	""
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	170, "Invalid equation label" 171, "Invalid equation

	expression"

Examples

```
Dim Var as String
SCPI.CALCulate.FP1.TRACe4.EQUation.TEXT = Var
Var = SCPI.CALCulate.FP1.TRACe4.EQUation.TEXT
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:EQUation:TEXT

:CALCulate:FP[1-1]:TRACe[1-4]:EQUation:TEXT?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:EQUation:TEXT Sample_Text"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:EQUation:TEXT ?"
30 ENTER 717;A$
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).FORMat.FREQuency

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).FORMat.FREQuency = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).FORMat.FREQuency

Description

This command sets/gets FP-frequency format. Init value changes by tr #, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	FP-frequency format
Data Type	Character string type (String)
Range	HZ|HZV|DHZ|PCT|PPM
Preset Value	HZ
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "HZ"
SCPI.CALCulate.FP1.TRACe4.FORMat.FREQuency = Var
Var = SCPI.CALCulate.FP1.TRACe4.FORMat.FREQuency
```

Equivalent Key

FP Menu: **Format** > **Frequency Format**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:FORMat:FREQuency {HZ|HZV|DHZ|PCT|PPM}

:CALCulate:FP[1-1]:TRACe[1-4]:FORMat:FREQuency?

Query Response

{HZ|HZV|DHZ|PCT|PPM} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:FORMat:FREQuency HZ"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:FORMat:FREQuency ?"
30 ENTER 717;A$
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.DOMain.X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.DOMain.X = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.DOMain.X

Description

This command sets/gets X analysis range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	X analysis range
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	BDMarker
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.FP1.TRACe4.FUNCtion.DOMain.X = Var
Var = SCPI.CALCulate.FP1.TRACe4.FUNCtion.DOMain.X
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.DOMain.X

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.DOMain.Y

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.LREGression.DATA

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.LREGression.MEMory

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.STATistics.DATA

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.TYPE

Equivalent Key

FP Menu: **Marker Function** > **Analysis Range (X)**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:FUNCtion:DOMain:X {FRANge|BDMarker}

:CALCulate:FP[1-1]:TRACe[1-4]:FUNCtion:DOMain:X?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:FUNCtion:DOMain:X FRANge"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:FUNCtion:DOMain:X ?"
30 ENTER 717;A$
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.DOMain.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.DOMain.Y = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.DOMain.Y

Description

This command sets/gets Y analysis range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Y analysis range
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	BDMarker
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.FP1.TRACe4.FUNCtion.DOMain.Y = Var
Var = SCPI.CALCulate.FP1.TRACe4.FUNCtion.DOMain.Y
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.DOMain.X

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.LREGression.DATA

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.LREGression.MEMory

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.STATistics.DATA

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.TYPE

Equivalent Key

FP Menu: **Marker Function** > **Analysis Range (Y)**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:FUNCtion:DOMain:Y {FRANge|BDMarker}

:CALCulate:FP[1-1]:TRACe[1-4]:FUNCtion:DOMain:Y?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:FUNCtion:DOMain:Y FRANge"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:FUNCtion:DOMain:Y ?"
30 ENTER 717;A$
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.LREGression.DATA_Q a, b

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.LREGression.DATA_Q a, b

Description

This command gets trace data linear regression, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*a*
Description	Trace data linear regression
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*b*
Description	Trace data linear regression
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
Var = SCPI.CALCulate.FP1.TRACe4.FUNCtion.LREGression.DATA_Q a, b
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.DOMain.X

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.DOMain.Y

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.LREGression.MEMory

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.STATistics.DATA

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.TYPE

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:FUNCtion:LREGression:DATA?

Query Response

{ - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:FUNCtion:LREGression:DATA ?"
30 ENTER 717;A$
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.LREGression.MEMory_Q a, b

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.LREGression.MEMory_Q a, b

Description

This command gets memory data linear regression, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*a*
Description	Memory data linear regression
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*b*
Description	Memory data linear regression
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
Var = SCPI.CALCulate.FP1.TRACe4.FUNCtion.LREGression.MEMory_Q a, b
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.DOMain.X

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.DOMain.Y

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.LREGression.DATA

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.STATistics.DATA

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.TYPE

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:FUNCtion:LREGression:MEMory?

Query Response

{ - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:FUNCtion:LREGression:MEMory ?"
30 ENTER 717;A$
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak

Description

This command gets trace data statistics, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*mean*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*std-dev*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*peak_to_peak*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-

Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
Var = SCPI.CALCulate.FP1.TRACe4.FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.DOMain.X

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.DOMain.Y

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.LREGression.DATA

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.LREGression.MEMory

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.TYPE

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:FUNCtion:STATistics:DATA?

Query Response

{ - - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:FUNCtion:STATistics:DATA ?"
30 ENTER 717;A$
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak

Description

This command gets memory data statistics, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*mean*
Description	Memory data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*std_dev*
Description	Memory data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*peak_to_peak*
Description	Memory data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-

Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
Var = SCPI.CALCulate.FP1.TRACe4.FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.DOMain.X

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.DOMain.Y

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.LREGression.DATA

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.LREGression.MEMory

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.STATistics.DATA

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.TYPE

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:FUNCtion:STATistics:MEMory?

Query Response

{ - - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:FUNCtion:STATistics:MEMory ?"
30 ENTER 717;A$
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.TYPE

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.TYPE = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.TYPE

Description

This command sets/gets Analysis type, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Analysis type
Data Type	Character string type (String)
Range	OFF|STATistics|LREGression
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.FP1.TRACe4.FUNCtion.TYPE = Var
Var = SCPI.CALCulate.FP1.TRACe4.FUNCtion.TYPE
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.DOMain.X

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.DOMain.Y

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.LREGression.DATA

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.LREGression.MEMory

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.STATistics.DATA

SCPI.CALCulate.FP(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory

Equivalent Key

FP Menu: **Marker Function** > **Analysis Type**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:FUNCtion:TYPE {OFF|STATistics|LREGression}

:CALCulate:FP[1-1]:TRACe[1-4]:FUNCtion:TYPE?

Query Response

{OFF|STATistics|LREGression} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:FUNCtion:TYPE OFF"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:FUNCtion:TYPE ?"
30 ENTER 717;A$
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).HOLD

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).HOLD = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).HOLD

Description

This command sets/gets Data hold, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Data hold
Data Type	Character string type (String)
Range	OFF|MAXimum|MINimum
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.FP1.TRACe4.HOLD = Var
Var = SCPI.CALCulate.FP1.TRACe4.HOLD
```

Equivalent Key

FP Menu: **Trace View** > **Data Hold**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:HOLD {OFF|MAXimum|MINimum}

:CALCulate:FP[1-1]:TRACe[1-4]:HOLD?

Query Response

{OFF|MAXimum|MINimum} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:HOLD OFF"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:HOLD ?"
30 ENTER 717;A$
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.FAIL

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.FAIL

Description

This command gets limit test pass/fail, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test result
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Boolean
Var= 1
Var = SCPI.CALCulate.FP1.TRACe4.LIMit.FAIL
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:LIMit:FAIL?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:LIMit:FAIL ?"
20 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.LOWer.LDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.LOWer.LDATa = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.LOWer.LDATa

Description

This command sets/gets expanded lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Expanded lower limit line
Data Type	Variant type Array (Range)
Range	1...1001
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1001) as Variant
SCPI.CALCulate.FP1.TRACe4.LIMit.LOWer.LDATa = Var
Var = SCPI.CALCulate.FP1.TRACe4.LIMit.LOWer.LDATa
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:LIMit:LOWer:LDATa {1...1001}

:CALCulate:FP[1-1]:TRACe[1-4]:LIMit:LOWer:LDATa?

Query Response

{1...1001} <newline><^END>

Example of use

```
10 Dim A(1:1001)
20 OUTPUT 717;":CALCulate:FP1:TRACe4:LIMit:LOWer:LDATa 1,1001"
30 OUTPUT 717;":CALCulate:FP1:TRACe4:LIMit:LOWer:LDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

Description

This command clears lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.FP1.TRACe4.LIMit.LOWer.SEGMent.CLEar

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:LIMit:LOWer:SEGMent:CLEar

Example of use

10 OUTPUT 717;":CALCulate:FP1:TRACe4:LIMit:LOWer:SEGMent:CLEar"

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

Description

This command sets/gets number of lower limit segments, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Lower limit segments
Data Type	Long integer type (Long)
Range	1 ~ 100
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.FP1.TRACe4.LIMit.LOWer.SEGMent.COUNt = Var
Var = SCPI.CALCulate.FP1.TRACe4.LIMit.LOWer.SEGMent.COUNt
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:LIMit:LOWer:SEGMent:COUNt {1 ~ 100}

:CALCulate:FP[1-1]:TRACe[1-4]:LIMit:LOWer:SEGMent:COUNt?

Query Response

{1 ~ 100} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:LIMit:LOWer:SEGMent:COUNt 1 "
20 OUTPUT 717;":CALCulate:FP1:TRACe4:LIMit:LOWer:SEGMent:COUNt ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Description

This command sets/gets lower limit line data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Lower limit data
Data Type	Variant type Array (Range)
Range	1...400
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 400) as Variant
SCPI.CALCulate.FP1.TRACe4.LIMit.LOWer.SEGMent.DATA = Var
Var = SCPI.CALCulate.FP1.TRACe4.LIMit.LOWer.SEGMent.DATA
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:LIMit:LOWer:SEGMent:DATA {1...400}

:CALCulate:FP[1-1]:TRACe[1-4]:LIMit:LOWer:SEGMent:DATA?

Query Response

{1...400} <newline><^END>

Example of use

```
10 Dim A(1:400)
20 OUTPUT 717;":CALCulate:FP1:TRACe4:LIMit:LOWer:SEGMent:DATA 1,400"
30 OUTPUT 717;":CALCulate:FP1:TRACe4:LIMit:LOWer:SEGMent:DATA ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.REPort.DATA

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.REPort.DATA

Description

This command gets pass/fail list, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test report
Data Type	Variant type Array (Range)
Range	1...1001
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1001) as Variant
Var = SCPI.CALCulate.FP1.TRACe4.LIMit.REPort.DATA
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:LIMit:REPort[:DATA]?

Query Response

{1...1001} <newline><^END>

Example of use

```
10 Dim A(1:1001)
20 OUTPUT 717;":CALCulate:FP1:TRACe4:LIMit:REPort[:DATA] ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.STATe = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.STATe

Description

This command sets/gets limit test on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.FP1.TRACe4.LIMit.STATe = Var
Var = SCPI.CALCulate.FP1.TRACe4.LIMit.STATe
```

Equivalent Key

FP Menu: **Display** > **Limit Test** > **Limit Test**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:LIMit[:STATe] {ON|OFF|1|0}

:CALCulate:FP[1-1]:TRACe[1-4]:LIMit[:STATe]?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:LIMit[:STATe] 1"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:LIMit[:STATe] ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.UPPer.LDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.UPPer.LDATa = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.UPPer.LDATa

Description

This command sets/gets expanded upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Upper limit line
Data Type	Variant type Array (Range)
Range	1...1001
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1001) as Variant
SCPI.CALCulate.FP1.TRACe4.LIMit.UPPer.LDATa = Var
Var = SCPI.CALCulate.FP1.TRACe4.LIMit.UPPer.LDATa
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:LIMit:UPPer:LDATa {1...1001}

:CALCulate:FP[1-1]:TRACe[1-4]:LIMit:UPPer:LDATa?

Query Response

{1...1001} <newline><^END>

Example of use

```
10 Dim A(1:1001)
20 OUTPUT 717;":CALCulate:FP1:TRACe4:LIMit:UPPer:LDATa 1,1001"
30 OUTPUT 717;":CALCulate:FP1:TRACe4:LIMit:UPPer:LDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

Description

This command clears upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.FP1.TRACe4.LIMit.UPPer.SEGMent.CLEar

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:LIMit:UPPer:SEGMent:CLEar

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:LIMit:UPPer:SEGMent:CLEar"
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Description

This command sets/gets number of upper limit segments, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Upper limit segment
Data Type	Long integer type (Long)
Range	1 ~ 100
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.FP1.TRACe4.LIMit.UPPer.SEGMent.COUNt = Var
Var = SCPI.CALCulate.FP1.TRACe4.LIMit.UPPer.SEGMent.COUNt
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:LIMit:UPPer:SEGMent:COUNt {1 ~ 100}

:CALCulate:FP[1-1]:TRACe[1-4]:LIMit:UPPer:SEGMent:COUNt?

Query Response

{1 ~ 100} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:LIMit:UPPer:SEGMent:COUNt 1 "
20 OUTPUT 717;":CALCulate:FP1:TRACe4:LIMit:UPPer:SEGMent:COUNt ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Description

This command sets/gets upper limit line data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Upper limit line data
Data Type	Variant type Array (Range)
Range	1...400
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 400) as Variant
SCPI.CALCulate.FP1.TRACe4.LIMit.UPPer.SEGMent.DATA = Var
Var = SCPI.CALCulate.FP1.TRACe4.LIMit.UPPer.SEGMent.DATA
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.FP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:LIMit:UPPer:SEGMent:DATA {1...400}

:CALCulate:FP[1-1]:TRACe[1-4]:LIMit:UPPer:SEGMent:DATA?

Query Response

{1...400} <newline><^END>

Example of use

```
10 Dim A(1:400)
20 OUTPUT 717;":CALCulate:FP1:TRACe4:LIMit:UPPer:SEGMent:DATA 1,400"
30 OUTPUT 717;":CALCulate:FP1:TRACe4:LIMit:UPPer:SEGMent:DATA ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).LINE.A

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).LINE.A = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).LINE.A

Description

This command sets/gets Parameter 'A' as line (Y = AX + B), for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Parameter 'A' in line (Y = AX + B)
Data Type	Double precision floating point type (Double)
Range	-500T ~ 500T
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -5000000000000
SCPI.CALCulate.FP1.TRACe4.LINE.A = Var
Var = SCPI.CALCulate.FP1.TRACe4.LINE.A
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).LINE.A

SCPI.CALCulate.FP(Ch).TRACe(Tr).LINE.B

SCPI.CALCulate.FP(Ch).TRACe(Tr).LINE.MEMory

Equivalent Key

FP Menu: **Trace View** > **Memory Trace** > **Line (Y = AX + B)** > **A**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:LINE:A {-500T ~ 500T}

:CALCulate:FP[1-1]:TRACe[1-4]:LINE:A?

Query Response

{-500T ~ 500T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:LINE:A -5000000000000"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:LINE:A ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).LINE.B

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).LINE.B = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).LINE.B

Description

This command sets/gets Parameter 'B' as line (Y = AX + B), for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Parameter 'B' in line (Y = AX + B)
Data Type	Double precision floating point type (Double)
Range	-500T ~ 500T
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -5000000000000
SCPI.CALCulate.FP1.TRACe4.LINE.B = Var
Var = SCPI.CALCulate.FP1.TRACe4.LINE.B
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).LINE.A

SCPI.CALCulate.FP(Ch).TRACe(Tr).LINE.MEMory

Equivalent Key

FP Menu: **Trace View** > **Memory Trace** > **Line (Y = AX + B)** > **B**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:LINE:B {-500T ~ 500T}

:CALCulate:FP[1-1]:TRACe[1-4]:LINE:B?

Query Response

{-500T ~ 500T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:LINE:B -5000000000000"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:LINE:B ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).LINE.MEMory

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).LINE.MEMory

Description

This command sets line (Y = AX + B) data to Memory trace, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.FP1.TRACe4.LINE.MEMory

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).LINE.A

SCPI.CALCulate.FP(Ch).TRACe(Tr).LINE.B

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:LINE:MEMory

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:LINE:MEMory"
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

Description

This command sets marker search LPEak, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.FP1.TRACe4.MARKer10.SEARch.EXECute.LPEak

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

FP Menu: **Marker Search** > **Peak** > **Search Left**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:MARKer[1-10]:SEARch:EXECute:LPEak

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:MARKer10:SEARch:EXECute:LPEak"
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

Description

This command sets marker search LTARget, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.FP1.TRACe4.MARKer10.SEARch.EXECute.LTARget

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

FP Menu: **Marker Search** > **Target** > **Search Left**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:MARKer[1-10]:SEARch:EXECute:LTARget

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:MARKer10:SEARch:EXECute:LTARget"
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

Description

This command sets marker search MAXimum, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.FP1.TRACe4.MARKer10.SEARch.EXECute.MAXimum

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

FP Menu: **Marker Search** > **Search Max**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:MARKer[1-10]:SEARch:EXECute:MAXimum

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:MARKer10:SEARch:EXECute:MAXimum"
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

Description

This command sets marker search MINimum, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.FP1.TRACe4.MARKer10.SEARch.EXECute.MINimum

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

FP Menu: **Marker Search** > **Search Min**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:MARKer[1-10]:SEARch:EXECute:MINimum

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:MARKer10:SEARch:EXECute:MINimum"
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

Description

This command sets marker search PEAK, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.FP1.TRACe4.MARKer10.SEARch.EXECute.PEAK

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

FP Menu: **Marker Search** > **Peak** > **Search Peak**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:MARKer[1-10]:SEARch:EXECute:PEAK

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:MARKer10:SEARch:EXECute:PEAK"
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

Description

This command sets marker search RPEak, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.FP1.TRACe4.MARKer10.SEARch.EXECute.RPEak

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

FP Menu: **Marker Search** > **Peak** > **Search Right**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:MARKer[1-10]:SEARch:EXECute:RPEak

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:MARKer10:SEARch:EXECute:RPEak"
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

Description

This command sets marker search RTARget, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.FP1.TRACe4.MARKer10.SEARch.EXECute.RTARget

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

FP Menu: **Marker Search** > **Target** > **Search Right**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:MARKer[1-10]:SEARch:EXECute:RTARget

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:MARKer10:SEARch:EXECute:RTARget"
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Description

This command sets marker search TARGet, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.FP1.TRACe4.MARKer10.SEARch.EXECute.TARGet

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

Equivalent Key

FP Menu: **Marker Search** > **Target** > **Search Target**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:MARKer[1-10]:SEARch:EXECute:TARGet

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:MARKer10:SEARch:EXECute:TARGet"
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion

Description

This command sets/gets marker-search-peak excursion value, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker-search-peak excursion value
Data Type	Double precision floating point type (Double)
Range	0 ~ 10G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.FP1.TRACe4.MARKer10.SEARch.PEAK.EXCursion = Var
Var = SCPI.CALCulate.FP1.TRACe4.MARKer10.SEARch.PEAK.EXCursion
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity

Equivalent Key

FP Menu: **Marker Search** > **Peak** > **Peak Excursion**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:MARKer[1-10]:SEARch:PEAK:EXCursion {0 ~ 10G}

:CALCulate:FP[1-1]:TRACe[1-4]:MARKer[1-10]:SEARch:PEAK:EXCursion?

Query Response

{0 ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:MARKer10:SEARch:PEAK:EXCursion 0"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:MARKer10:SEARch:PEAK:EXCursion ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity

Description

This command sets/gets marker-search-peak polarity type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker-search-peak polarity type
Data Type	Character string type (String)
Range	POSitive|NEGative|BOTH
Preset Value	POSitive
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "POSitive"
SCPI.CALCulate.FP1.TRACe4.MARKer10.SEARch.PEAK.POLarity = Var
Var = SCPI.CALCulate.FP1.TRACe4.MARKer10.SEARch.PEAK.POLarity
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion

Equivalent Key

FP Menu: **Marker Search** > **Peak** > **Peak Polarity**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:MARKer[1-10]:SEARch:PEAK:POLarity {POSitive|NEGative|BOTH}

:CALCulate:FP[1-1]:TRACe[1-4]:MARKer[1-10]:SEARch:PEAK:POLarity?

Query Response

{POSitive|NEGative|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:MARKer10:SEARch:PEAK:POLarity POSitive"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:MARKer10:SEARch:PEAK:POLarity ?"
30 ENTER 717;A$
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition

Description

This command sets/gets marker-target transition type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker target transition type
Data Type	Character string type (String)
Range	POSitive|NEGative|BOTH
Preset Value	BOTH
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as String
Var= "POSitive"
SCPI.CALCulate.FP1.TRACe4.MARKer10.SEARch.TARGet.TRANsition = Var
Var = SCPI.CALCulate.FP1.TRACe4.MARKer10.SEARch.TARGet.TRANsition
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y

Equivalent Key

FP Menu: **Marker Search** > **Target** > **Target Transition**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:MARKer[1-10]:SEARch:TARGet:TRANsition {POSitive|NEGative|BOTH}

:CALCulate:FP[1-1]:TRACe[1-4]:MARKer[1-10]:SEARch:TARGet:TRANsition?

Query Response

{POSitive|NEGative|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:MARKer10:SEARch:TARGet:TRANsition POSitive"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:MARKer10:SEARch:TARGet:TRANsition ?"
30 ENTER 717;A$
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y

Description

This command sets/gets marker target value, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker target value
Data Type	Double precision floating point type (Double)
Range	-10G ~ 10G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.FP1.TRACe4.MARKer10.SEARch.TARGet.Y = Var
Var = SCPI.CALCulate.FP1.TRACe4.MARKer10.SEARch.TARGet.Y
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition

Equivalent Key

FP Menu: **Marker Search** > **Target** > **Target Value**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:MARKer[1-10]:SEARch:TARGet:Y {-10G ~ 10G}

:CALCulate:FP[1-1]:TRACe[1-4]:MARKer[1-10]:SEARch:TARGet:Y?

Query Response

{-10G ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:MARKer10:SEARch:TARGet:Y -10000000000"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:MARKer10:SEARch:TARGet:Y ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE

Description

This command sets/gets search tracking type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search tracking type
Data Type	Character string type (String)
Range	OFF|MAXimum|MINimum|PEAK|TARGet
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.FP1.TRACe4.MARKer10.SEARch.TRACking.TYPE = Var
Var = SCPI.CALCulate.FP1.TRACe4.MARKer10.SEARch.TRACking.TYPE
```

Equivalent Key

FP Menu: **Marker Search** > **Tracking**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:MARKer[1-10]:SEARch:TRACking:TYPE {OFF|MAXimum|MINimum|PEAK|TARGet}

:CALCulate:FP[1-1]:TRACe[1-4]:MARKer[1-10]:SEARch:TRACking:TYPE?

Query Response

{OFF|MAXimum|MINimum|PEAK|TARGet} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:MARKer10:SEARch:TRACking:TYPE OFF"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:MARKer10:SEARch:TRACking:TYPE ?"
30 ENTER 717;A$
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).STATe = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).STATe

Description

This command sets/gets marker visible on/off, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker visible state
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.FP1.TRACe4.MARKer10.STATe = Var
Var = SCPI.CALCulate.FP1.TRACe4.MARKer10.STATe
```

Equivalent Key

FP Menu: **Marker** > **Clear Marker Menu** > **Marker 1**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:MARKer[1-10]:STATe {ON|OFF|1|0}

:CALCulate:FP[1-1]:TRACe[1-4]:MARKer[1-10]:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:MARKer10:STATe 1"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:MARKer10:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).X = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).X

Description

This command sets/gets marker x position., for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker X position
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
SCPI.CALCulate.FP1.TRACe4.MARKer10.X = Var
Var = SCPI.CALCulate.FP1.TRACe4.MARKer10.X
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).Y

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:MARKer[1-10]:X

:CALCulate:FP[1-1]:TRACe[1-4]:MARKer[1-10]:X?

Query Response

{Double} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:MARKer10:X 0"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:MARKer10:X ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).Y

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).Y

Description

This command gets marker y position, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker Y position
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var = SCPI.CALCulate.FP1.TRACe4.MARKer10.Y
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).MARKer(Mk).X

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:MARKer[1-10]:Y?

Query Response

{Double} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:MARKer10:Y ?"
20 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).MATH.FUNCtion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).MATH.FUNCtion = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).MATH.FUNCtion

Description

This command sets/gets Select math operation type, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Math function operation
Data Type	Character string type (String)
Range	NORMal|SUBTract|DIVide|ADD|MULTiply|USER
Preset Value	NORMal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NORMal"
SCPI.CALCulate.FP1.TRACe4.MATH.FUNCtion = Var
Var = SCPI.CALCulate.FP1.TRACe4.MATH.FUNCtion
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).MATH.FUNCtion

SCPI.CALCulate.FP(Ch).TRACe(Tr).MATH.MEMorize

SCPI.CALCulate.FP(Ch).TRACe(Tr).MATH.OFFSet

Equivalent Key

FP Menu: **Trace View** > **Data Math**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:MATH:FUNCtion
{NORMal|SUBTract|DIVide|ADD|MULTiply|USER}

:CALCulate:FP[1-1]:TRACe[1-4]:MATH:FUNCtion?

Query Response

{NORMal|SUBTract|DIVide|ADD|MULTiply|USER} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:MATH:FUNCtion NORMal"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:MATH:FUNCtion ?"
30 ENTER 717;A$
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).MATH.MEMorize

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).MATH.MEMorize

Description

This command copies data to memory, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.FP1.TRACe4.MATH.MEMorize

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).MATH.FUNCtion

SCPI.CALCulate.FP(Ch).TRACe(Tr).MATH.OFFSet

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:MATH:MEMorize

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:MATH:MEMorize"
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).MATH.OFFSet

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).MATH.OFFSet = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).MATH.OFFSet

Description

This command sets/gets Trace Offset, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace offset
Data Type	Double precision floating point type (Double)
Range	-500G ~ 500G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -500000000000
SCPI.CALCulate.FP1.TRACe4.MATH.OFFSet = Var
Var = SCPI.CALCulate.FP1.TRACe4.MATH.OFFSet
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).MATH.FUNCtion

SCPI.CALCulate.FP(Ch).TRACe(Tr).MATH.MEMorize

Equivalent Key

FP Menu: **Trace View** > **Offset**

FP Menu: **Trace View** > **Marker** > **-Offset**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:MATH:OFFSet {-500G ~ 500G}

:CALCulate:FP[1-1]:TRACe[1-4]:MATH:OFFSet?

Query Response

{-500G ~ 500G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:MATH:OFFSet -500000000000"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:MATH:OFFSet ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).PARameter

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).PARameter

Description

This command gets FP measurement type. Init value changes by tr #., for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	FP measurement type
Data Type	Character string type (String)
Range	FREQuency|POWer|CURRent
Preset Value	FREQuency
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as String
Var= "FREQuency"
Var = SCPI.CALCulate.FP1.TRACe4.PARameter
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:PARameter?

Query Response

{FREQuency|POWer|CURRent} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:PARameter ?"
20 ENTER 717;A$
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).REFerence.FREQuency

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).REFerence.FREQuency = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).REFerence.FREQuency

Description

This command sets/gets frequency reference for DHZ, PCT, PPM, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Frequency reference for DHZ, PCT, PPM
Data Type	Double precision floating point type (Double)
Range	-500G ~ 500G
Preset Value	0
Unit	Hz
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -500000000000
SCPI.CALCulate.FP1.TRACe4.REFerence.FREQuency = Var
Var = SCPI.CALCulate.FP1.TRACe4.REFerence.FREQuency
```

Equivalent Key

FP Menu: **Format** > **Frequency Reference**

FP Menu: **Format** > **Marker** > **Frequency Reference**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:REFerence:FREQuency {-500G ~ 500G}

:CALCulate:FP[1-1]:TRACe[1-4]:REFerence:FREQuency?

Query Response

{-500G ~ 500G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:REFerence:FREQuency -500000000000"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:REFerence:FREQuency ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).SAPerture

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).SAPerture = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).SAPerture

Description

This command sets/gets Sensitivity Aperture, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Sensitivity Aperture
Data Type	Double precision floating point type (Double)
Range	100m ~ 20
Preset Value	1
Unit	%
Resolution	100m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0.1
SCPI.CALCulate.FP1.TRACe4.SAPerture = Var
Var = SCPI.CALCulate.FP1.TRACe4.SAPerture
```

Related Objects

Equivalent Key

FP Menu: **Format** > **Sensitivity Aperture**

Equivalent SCPI Command

Syntax

```
:CALCulate:FP[1-1]:TRACe[1-4]:SAPerture {100m ~ 20}
:CALCulate:FP[1-1]:TRACe[1-4]:SAPerture?
```

Query Response

```
{100m ~ 20} <newline><^END>
```

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:SAPerture 0.1"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:SAPerture ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).SMOothing.APERture

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).SMOothing.APERture = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).SMOothing.APERture

Description

This command sets/gets Smoothing aperture, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Smoothing aperture
Data Type	Double precision floating point type (Double)
Range	50m ~ 25
Preset Value	1.5
Unit	%
Resolution	10m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0.05
SCPI.CALCulate.FP1.TRACe4.SMOothing.APERture = Var
Var = SCPI.CALCulate.FP1.TRACe4.SMOothing.APERture
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).SMOothing.STATe

Equivalent Key

FP Menu: **Trace View** > **Aperture**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:SMOothing:APERture {50m ~ 25}

:CALCulate:FP[1-1]:TRACe[1-4]:SMOothing:APERture?

Query Response

{50m ~ 25} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:SMOothing:APERture 0.05"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:SMOothing:APERture ?"
30 ENTER 717;A
```

SCPI.CALCulate.FP(Ch).TRACe(Tr).SMOothing.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.FP(Ch).TRACe(Tr).SMOothing.STATe = *Value*

Value = SCPI.CALCulate.FP(Ch).TRACe(Tr).SMOothing.STATe

Description

This command sets/gets Smoothing on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Smoothing on/off
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.FP1.TRACe4.SMOothing.STATe = Var
Var = SCPI.CALCulate.FP1.TRACe4.SMOothing.STATe
```

Related Objects

SCPI.CALCulate.FP(Ch).TRACe(Tr).SMOothing.APERture

Equivalent Key

FP Menu: **Trace View** > **Smoothing**

Equivalent SCPI Command

Syntax

:CALCulate:FP[1-1]:TRACe[1-4]:SMOothing:STATe {ON|OFF|1|0}

:CALCulate:FP[1-1]:TRACe[1-4]:SMOothing:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:FP1:TRACe4:SMOothing:STATe 1"
20 OUTPUT 717;":CALCulate:FP1:TRACe4:SMOothing:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).ALLTrace.LIMit.FAIL

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PN(Ch).ALLTrace.LIMit.FAIL

Description

This command gets limit test pass/fail, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Result of limit test
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Boolean
Var= 1
Var = SCPI.CALCulate.PN1.ALLTrace.LIMit.FAIL
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:ALLTrace:LIMit:FAIL?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:ALLTrace:LIMit:FAIL ?"
20 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.COUPle.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.COUPle.STATe = *Value*

Value = SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.COUPle.STATe

Description

This command sets/gets marker couple on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker Coupling Function state (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.PN1.ALLTrace.MARKer.COUPle.STATe = Var
Var = SCPI.CALCulate.PN1.ALLTrace.MARKer.COUPle.STATe
```

Related Objects

SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.DISCrete.STATe

SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.REFerence.NUMBer

SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.REFerence.STATe

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:ALLTrace:MARKer:COUPle:STATe {ON|OFF|1|0}

:CALCulate:PN[1-1]:ALLTrace:MARKer:COUPle:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:ALLTrace:MARKer:COUPle:STATe 1"
20 OUTPUT 717;":CALCulate:PN1:ALLTrace:MARKer:COUPle:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.DISCrete.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.DISCrete.STATe = *Value*

Value = SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.DISCrete.STATe

Description

This command sets/gets marker discrete on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Enables/Disables marker discrete
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.PN1.ALLTrace.MARKer.DISCrete.STATe = Var
Var = SCPI.CALCulate.PN1.ALLTrace.MARKer.DISCrete.STATe
```

Related Objects

SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.COUPle.STATe

SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.REFerence.NUMBer

SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.REFerence.STATe

Equivalent Key

PN Menu: **Marker** > **More Functions** > **Discrete**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:ALLTrace:MARKer:DISCrete:STATe {ON|OFF|1|0}

:CALCulate:PN[1-1]:ALLTrace:MARKer:DISCrete:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:ALLTrace:MARKer:DISCrete:STATe 1"
20 OUTPUT 717;":CALCulate:PN1:ALLTrace:MARKer:DISCrete:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.REFerence.NUMBer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.REFerence.NUMBer = *Value*

Value = SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.REFerence.NUMBer

Description

This command sets/gets reference marker number., for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker reference number
Data Type	Long integer type (Long)
Range	1 ~ 10
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.PN1.ALLTrace.MARKer.REFerence.NUMBer = Var
Var = SCPI.CALCulate.PN1.ALLTrace.MARKer.REFerence.NUMBer
```

Related Objects

SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.COUPle.STATe

SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.DISCrete.STATe

SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.REFerence.STATe

Equivalent Key

PN Menu: **Marker** > **More Functions** > **Ref Marker**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:ALLTrace:MARKer:REFerence:NUMBer {1 ~ 10}

:CALCulate:PN[1-1]:ALLTrace:MARKer:REFerence:NUMBer?

Query Response

{1 ~ 10} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:ALLTrace:MARKer:REFerence:NUMBer 1 "
20 OUTPUT 717;":CALCulate:PN1:ALLTrace:MARKer:REFerence:NUMBer ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.REFerence.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.REFerence.STATe = *Value*

Value = SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.REFerence.STATe

Description

This command sets/gets delta marker mode on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Delta marker mode (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.PN1.ALLTrace.MARKer.REFerence.STATe = Var
Var = SCPI.CALCulate.PN1.ALLTrace.MARKer.REFerence.STATe
```

Related Objects

SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.COUPle.STATe

SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.DISCrete.STATe

SCPI.CALCulate.PN(Ch).ALLTrace.MARKer.REFerence.NUMBer

Equivalent Key

PN Menu: **Marker** > **More Functions** > **Ref Marker Mode**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:ALLTrace:MARKer:REFerence:STATe {ON|OFF|1|0}

:CALCulate:PN[1-1]:ALLTrace:MARKer:REFerence:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:ALLTrace:MARKer:REFerence:STATe 1"
20 OUTPUT 717;":CALCulate:PN1:ALLTrace:MARKer:REFerence:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).DATA.CARRier

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).DATA.CARRier = *Value*

Value = SCPI.CALCulate.PN(Ch).DATA.CARRier

Description

This command sets/gets carrier data: freq, power, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Carrier data: Freq & Power
Data Type	Variant type Array (Range)
Range	1...2
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 2) as Variant
SCPI.CALCulate.PN1.DATA.CARRier = Var
Var = SCPI.CALCulate.PN1.DATA.CARRier
```

Related Objects

SCPI.CALCulate.PN(Ch).DATA.PDATa

SCPI.CALCulate.PN(Ch).DATA.RDATa

SCPI.CALCulate.PN(Ch).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:DATA:CARRier {1...2}

:CALCulate:PN[1-1]:DATA:CARRier?

Query Response

{1...2} <newline><^END>

Example of use

```
10 Dim A(1:2)
20 OUTPUT 717;":CALCulate:PN1:DATA:CARRier 1,2"
30 OUTPUT 717;":CALCulate:PN1:DATA:CARRier ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PN(Ch).DATA.PDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).DATA.PDATa = *Value*

Value = SCPI.CALCulate.PN(Ch).DATA.PDATa

Description

This command sets/gets measurement raw power data, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Measurement raw power data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.PN1.DATA.PDATa = Var
Var = SCPI.CALCulate.PN1.DATA.PDATa
```

Related Objects

SCPI.CALCulate.PN(Ch).DATA.CARRier

SCPI.CALCulate.PN(Ch).DATA.RDATa

SCPI.CALCulate.PN(Ch).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:DATA:PDATa {1...1601}

:CALCulate:PN[1-1]:DATA:PDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PN1:DATA:PDATa 1,1601"
30 OUTPUT 717;":CALCulate:PN1:DATA:PDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PN(Ch).DATA.RDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).DATA.RDATa = *Value*

Value = SCPI.CALCulate.PN(Ch).DATA.RDATa

Description

This command sets/gets measurement raw data, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Measurement raw data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.PN1.DATA.RDATa = Var
Var = SCPI.CALCulate.PN1.DATA.RDATa
```

Related Objects

SCPI.CALCulate.PN(Ch).DATA.CARRier

SCPI.CALCulate.PN(Ch).DATA.PDATa

SCPI.CALCulate.PN(Ch).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:DATA:RDATa {1...1601}

:CALCulate:PN[1-1]:DATA:RDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PN1:DATA:RDATa 1,1601"
30 OUTPUT 717;":CALCulate:PN1:DATA:RDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PN(Ch).DATA.XDATa

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PN(Ch).DATA.XDATa

Description

This command gets X axis data, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	X axis data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
Var = SCPI.CALCulate.PN1.DATA.XDATa
```

Related Objects

SCPI.CALCulate.PN(Ch).DATA.CARRier

SCPI.CALCulate.PN(Ch).DATA.PDATa

SCPI.CALCulate.PN(Ch).DATA.RDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:DATA:XDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PN1:DATA:XDATa ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).ALLMarker.ACTive

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).ALLMarker.ACTive = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).ALLMarker.ACTive

Description

This command sets/gets active marker, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Active marker
Data Type	Long integer type (Long)
Range	1 ~ 10
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.PN1.TRACe1.ALLMarker.ACTive = Var
Var = SCPI.CALCulate.PN1.TRACe1.ALLMarker.ACTive
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:ALLMarker:ACTive {1 ~ 10}

:CALCulate:PN[1-1]:TRACe[1-1]:ALLMarker:ACTive?

Query Response

{1 ~ 10} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:ALLMarker:ACTive 1 "
20 OUTPUT 717;":CALCulate:PN1:TRACe1:ALLMarker:ACTive ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

Description

This command sets/gets marker search X range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search X range source
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	FRANge
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.PN1.TRACe1.ALLMarker.SEARch.DOMain.X = Var
Var = SCPI.CALCulate.PN1.TRACe1.ALLMarker.SEARch.DOMain.X
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

SCPI.CALCulate.PN(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

SCPI.CALCulate.PN(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Equivalent Key

PN Menu: **Marker Search** > **Search Range (X)**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:ALLMarker:SEARch:DOMain:X {FRANge|BDMarker}

:CALCulate:PN[1-1]:TRACe[1-1]:ALLMarker:SEARch:DOMain:X?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:ALLMarker:SEARch:DOMain:X FRANge"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:ALLMarker:SEARch:DOMain:X ?"
30 ENTER 717;A$
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Description

This command sets/gets marker search Y range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search Y range
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	FRANge
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.PN1.TRACe1.ALLMarker.SEARch.DOMain.Y = Var
Var = SCPI.CALCulate.PN1.TRACe1.ALLMarker.SEARch.DOMain.Y
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

SCPI.CALCulate.PN(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Equivalent Key

PN Menu: **Marker Search** > **Search Range (Y)**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:ALLMarker:SEARch:DOMain:Y {FRANge|BDMarker}

:CALCulate:PN[1-1]:TRACe[1-1]:ALLMarker:SEARch:DOMain:Y?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:ALLMarker:SEARch:DOMain:Y FRANge"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:ALLMarker:SEARch:DOMain:Y ?"
30 ENTER 717;A$
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Description

This command sets search peak all, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PN1.TRACe1.ALLMarker.SEARch.PEAK

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

SCPI.CALCulate.PN(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Equivalent Key

PN Menu: **Marker Search** > **Peak** > **Search Peak All**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:ALLMarker:SEARch:PEAK

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:ALLMarker:SEARch:PEAK"
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).ALLMarker.SEARch.SPURious

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).ALLMarker.SEARch.SPURious

Description

This command sets search spurious all, for the selected trace *Tr* of the selected channel *Ch*. You can display up to 10 markers.

NOTE This function is allowed to use when the spurious display is power.

Examples

SCPI.CALCulate.PN1.TRACe1.ALLMarker.SEARch.SPURious

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

Equivalent Key

PN Menu: **Marker Search** > **Spurious** > **Search Spurious All**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:ALLMarker:SEARch:SPURious

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:ALLMarker:SEARch:SPURious"
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.CENTer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.CENTer = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.CENTer

Description

This command sets/gets band marker X center, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X center
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.PN1.TRACe1.BDMarker.X.CENTer = Var
Var = SCPI.CALCulate.PN1.TRACe1.BDMarker.X.CENTer
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

PN Menu: **Marker Search** > **Band Marker X** > **Center**

PN Menu: **Marker Function** > **Band Marker X** > **Center**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:BDMarker:X:CENTer {-1T ~ 1T}

:CALCulate:PN[1-1]:TRACe[1-1]:BDMarker:X:CENTer?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:BDMarker:X:CENTer -10000000000"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:BDMarker:X:CENTer ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.SPAN

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.SPAN = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.SPAN

Description

This command sets/gets band marker X span, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X span
Data Type	Double precision floating point type (Double)
Range	0 ~ 2T
Preset Value	2T
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.PN1.TRACe1.BDMarker.X.SPAN = Var
Var = SCPI.CALCulate.PN1.TRACe1.BDMarker.X.SPAN
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

PN Menu: **Marker Search** > **Band Marker X** > **Span**

PN Menu: **Marker Function** > **Band Marker X** > **Span**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:BDMarker:X:SPAN {0 ~ 2T}

:CALCulate:PN[1-1]:TRACe[1-1]:BDMarker:X:SPAN?

Query Response

{0 ~ 2T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:BDMarker:X:SPAN 0"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:BDMarker:X:SPAN ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.STARt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.STARt = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.STARt

Description

This command sets/gets band marker X start, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X start
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	-1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.PN1.TRACe1.BDMarker.X.STARt = Var
Var = SCPI.CALCulate.PN1.TRACe1.BDMarker.X.STARt
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

PN Menu: **Marker Search** > **Band Marker X** > **Start**

PN Menu: **Marker Function** > **Band Marker X** > **Start**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:BDMarker:X:STARt {-1T ~ 1T}

:CALCulate:PN[1-1]:TRACe[1-1]:BDMarker:X:STARt?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:BDMarker:X:STARt -10000000000"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:BDMarker:X:STARt ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.STATe = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.STATe

Description

This command sets/gets band marker visible on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker visibility (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.PN1.TRACe1.BDMarker.X.STATe = Var
Var = SCPI.CALCulate.PN1.TRACe1.BDMarker.X.STATe
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

PN Menu: **Marker Search** > **Band Marker X** > **Band Marker X**

PN Menu: **Marker Function** > **Band Marker X** > **Band Marker X**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:BDMarker:X:STATe {ON|OFF|1|0}

:CALCulate:PN[1-1]:TRACe[1-1]:BDMarker:X:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:BDMarker:X:STATe 1"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:BDMarker:X:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.STOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.STOP = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.STOP

Description

This command sets/gets band marker X stop, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X stop
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.PN1.TRACe1.BDMarker.X.STOP = Var
Var = SCPI.CALCulate.PN1.TRACe1.BDMarker.X.STOP
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.X.STATe

Equivalent Key

PN Menu: **Marker Search** > **Band Marker X** > **Stop**

PN Menu: **Marker Function** > **Band Marker X** > **Stop**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:BDMarker:X:STOP {-1T ~ 1T}

:CALCulate:PN[1-1]:TRACe[1-1]:BDMarker:X:STOP?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:BDMarker:X:STOP -10000000000"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:BDMarker:X:STOP ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.CENTer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.CENTer = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.CENTer

Description

This command sets/gets band marker Y center, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y center
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.PN1.TRACe1.BDMarker.Y.CENTer = Var
Var = SCPI.CALCulate.PN1.TRACe1.BDMarker.Y.CENTer
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

PN Menu: **Marker Search** > **Band Marker Y** > **Center**

PN Menu: **Marker Function** > **Band Marker Y** > **Center**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:BDMarker:Y:CENTer {-1T ~ 1T}

:CALCulate:PN[1-1]:TRACe[1-1]:BDMarker:Y:CENTer?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:BDMarker:Y:CENTer -10000000000"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:BDMarker:Y:CENTer ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.SPAN

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.SPAN = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.SPAN

Description

This command sets/gets band marker Y span, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y span
Data Type	Double precision floating point type (Double)
Range	0 ~ 2T
Preset Value	2T
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.PN1.TRACe1.BDMarker.Y.SPAN = Var
Var = SCPI.CALCulate.PN1.TRACe1.BDMarker.Y.SPAN
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

PN Menu: **Marker Search** > **Band Marker Y** > **Span**

PN Menu: **Marker Function** > **Band Marker Y** > **Span**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:BDMarker:Y:SPAN {0 ~ 2T}

:CALCulate:PN[1-1]:TRACe[1-1]:BDMarker:Y:SPAN?

Query Response

{0 ~ 2T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:BDMarker:Y:SPAN 0"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:BDMarker:Y:SPAN ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.STARt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.STARt = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.STARt

Description

This command sets/gets band marker Y start, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y start
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	-1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.PN1.TRACe1.BDMarker.Y.STARt = Var
Var = SCPI.CALCulate.PN1.TRACe1.BDMarker.Y.STARt
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

PN Menu: **Marker Search** > **Band Marker Y** > **Start**

PN Menu: **Marker Function** > **Band Marker Y** > **Start**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:BDMarker:Y:STARt {-1T ~ 1T}

:CALCulate:PN[1-1]:TRACe[1-1]:BDMarker:Y:STARt?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:BDMarker:Y:STARt -10000000000"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:BDMarker:Y:STARt ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.STATe = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.STATe

Description

This command sets/gets band marker visible on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y state
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.PN1.TRACe1.BDMarker.Y.STATe = Var
Var = SCPI.CALCulate.PN1.TRACe1.BDMarker.Y.STATe
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

PN Menu: **Marker Search** > **Band Marker Y** > **Band Marker Y**

PN Menu: **Marker Function** > **Band Marker Y** > **Band Marker Y**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:BDMarker:Y:STATe {ON|OFF|1|0}

:CALCulate:PN[1-1]:TRACe[1-1]:BDMarker:Y:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:BDMarker:Y:STATe 1"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:BDMarker:Y:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.STOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.STOP = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.STOP

Description

This command sets/gets band marker Y stop, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y stop
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.PN1.TRACe1.BDMarker.Y.STOP = Var
Var = SCPI.CALCulate.PN1.TRACe1.BDMarker.Y.STOP
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.PN(Ch).TRACe(Tr).BDMarker.Y.STATe

Equivalent Key

PN Menu: **Marker Search** > **Band Marker Y** > **Stop**

PN Menu: **Marker Function** > **Band Marker Y** > **Stop**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:BDMarker:Y:STOP {-1T ~ 1T}

:CALCulate:PN[1-1]:TRACe[1-1]:BDMarker:Y:STOP?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:BDMarker:Y:STOP -10000000000"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:BDMarker:Y:STOP ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.COPY

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.COPY = *Value*

Description

This command copied data to user trace, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Copies data to user trace
Data Type	Long integer type (Long)
Range	1 ~ 8
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.PN1.TRACe1.DATA.COPY = Var
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:DATA:COPY {1 ~ 8}

Query Response

{1 ~ 8} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:DATA:COPY 1 "
20 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.FDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.FDATa = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.FDATa

Description

This command sets/gets formatted trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Formatted trace data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.PN1.TRACe1.DATA.FDATa = Var
Var = SCPI.CALCulate.PN1.TRACe1.DATA.FDATa
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:DATA:FDATa {1...1601}

:CALCulate:PN[1-1]:TRACe[1-1]:DATA:FDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PN1:TRACe1:DATA:FDATa 1,1601"
30 OUTPUT 717;":CALCulate:PN1:TRACe1:DATA:FDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.FMEMory

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.FMEMory = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.FMEMory

Description

This command sets/gets formatted memory data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Formatted memory data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.PN1.TRACe1.DATA.FMEMory = Var
Var = SCPI.CALCulate.PN1.TRACe1.DATA.FMEMory
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:DATA:FMEMory {1...1601}

:CALCulate:PN[1-1]:TRACe[1-1]:DATA:FMEMory?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PN1:TRACe1:DATA:FMEMory 1,1601"
30 OUTPUT 717;":CALCulate:PN1:TRACe1:DATA:FMEMory ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.PDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.PDATa = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.PDATa

Description

This command sets/gets dBc data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	dBC data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.PN1.TRACe1.DATA.PDATa = Var
Var = SCPI.CALCulate.PN1.TRACe1.DATA.PDATa
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:DATA:PDATa {1...1601}

:CALCulate:PN[1-1]:TRACe[1-1]:DATA:PDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PN1:TRACe1:DATA:PDATa 1,1601"
30 OUTPUT 717;":CALCulate:PN1:TRACe1:DATA:PDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.PMEMory

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.PMEMory = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.PMEMory

Description

This command sets/gets dBc memory data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	dBc memory data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.PN1.TRACe1.DATA.PMEMory = Var
Var = SCPI.CALCulate.PN1.TRACe1.DATA.PMEMory
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:DATA:PMEMory {1...1601}

:CALCulate:PN[1-1]:TRACe[1-1]:DATA:PMEMory?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PN1:TRACe1:DATA:PMEMory 1,1601"
30 OUTPUT 717;":CALCulate:PN1:TRACe1:DATA:PMEMory ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.SDATa

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.SDATa

Description

This command gets spurious data (0 or 1), for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
Var = SCPI.CALCulate.PN1.TRACe1.DATA.SDATa
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:DATA:SDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PN1:TRACe1:DATA:SDATa ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.SMEMory

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.SMEMory

Description

This command gets spurious memory data (0 or 1), for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious memory data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
Var = SCPI.CALCulate.PN1.TRACe1.DATA.SMEMory
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:DATA:SMEMory?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PN1:TRACe1:DATA:SMEMory ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.UDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.UDATa = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.UDATa

Description

This command sets/gets unformatted trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Unformatted trace data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.PN1.TRACe1.DATA.UDATa = Var
Var = SCPI.CALCulate.PN1.TRACe1.DATA.UDATa
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:DATA:UDATa {1...1601}

:CALCulate:PN[1-1]:TRACe[1-1]:DATA:UDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PN1:TRACe1:DATA:UDATa 1,1601"
30 OUTPUT 717;":CALCulate:PN1:TRACe1:DATA:UDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.UMEMory

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.UMEMory = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.UMEMory

Description

This command sets/gets unformatted memory data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Unformatted memory data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.PN1.TRACe1.DATA.UMEMory = Var
Var = SCPI.CALCulate.PN1.TRACe1.DATA.UMEMory
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.PN(Ch).TRACe(Tr).DATA.UDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:DATA:UMEMory {1...1601}

:CALCulate:PN[1-1]:TRACe[1-1]:DATA:UMEMory?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PN1:TRACe1:DATA:UMEMory 1,1601"
30 OUTPUT 717;":CALCulate:PN1:TRACe1:DATA:UMEMory ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).EQUation.TEXT

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).EQUation.TEXT = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).EQUation.TEXT

Description

This command sets/gets equation string, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Equation string
Data Type	Character string type (String)
Range	254 chars
Preset Value	""
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	170, "Invalid equation label" 171, "Invalid equation

	expression"

Examples

```
Dim Var as String
SCPI.CALCulate.PN1.TRACe1.EQUation.TEXT = Var
Var = SCPI.CALCulate.PN1.TRACe1.EQUation.TEXT
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:EQUation:TEXT

:CALCulate:PN[1-1]:TRACe[1-1]:EQUation:TEXT?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:EQUation:TEXT Sample_Text"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:EQUation:TEXT ?"
30 ENTER 717;A$
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.AVARiance.DATA_Q avg_time, fcutoff, allan_var, jitter

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.AVARiance.DATA_Q avg_time, fcutoff, allan_var, jitter

Description

This command gets Allan Variance (data trace), for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*avg_time*
Description	Average time
Data Type	Double precision floating point type (Double)
Range	1n ~ 1k
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*fcutoff*
Description	Frequency cutoff
Data Type	Double precision floating point type (Double)
Range	0 ~ 1G
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*allan_var*
Description	Allan variance
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-

Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*jitter*
Description	Jitter
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim avg_time As Double
Dim fcutoff As Double
Dim allan_var As Double
Dim jitter As Double
```

```
Dim var As Variant
avg_time = 0.001 'Tau = 0.001 sec
fcutoff = 10000 '10 kHz cutoff frequency
SCPI.CALCulate.PN.TRACe.FUNCtion.AVARiance.DATA_Q avg_time, fcutoff, allan_var, jitter
MsgBox "Allan Variance = " & Str$(allan_var) & " " & "Jitter(rms) = " & Str$(jitter) & " (sec)"
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.AVARiance.MEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:FUNCtion:AVARiance:DATA?

Query Response

{1n ~ 1k 0 ~ 1G - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:FUNCtion:AVARiance:DATA ?"
30 ENTER 717;A$
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.AVARiance.MEMory_Q avg_time, fcutoff, allan_var, jitter

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.AVARiance.MEMory_Q avg_time, fcutoff, allan_var, jitter

Description

This command gets Allan Variance (memory trace), for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*avg_time*
Description	Average time
Data Type	Double precision floating point type (Double)
Range	1n ~ 1k
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*fcutoff*
Description	Frequency cutoff
Data Type	Double precision floating point type (Double)
Range	0 ~ 1G
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*allan_var*
Description	Allan variance
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-

Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*jitter*
Description	Jitter
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
Var = SCPI.CALCulate.PN1.TRACe1.FUNCtion.AVARiance.MEMory_Q avg_time, fcutoff, allan_var, jitter
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.AVARiance.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:FUNCtion:AVARiance:MEMory?

Query Response

{1n ~ 1k 0 ~ 1G - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:FUNCtion:AVARiance:MEMory ?"
30 ENTER 717;A$
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.DOMain.X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.DOMain.X = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.DOMain.X

Description

This command sets/gets X analysis range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	X analysis range
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	BDMarker
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.PN1.TRACe1.FUNCtion.DOMain.X = Var
Var = SCPI.CALCulate.PN1.TRACe1.FUNCtion.DOMain.X
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.DOMain.Y

Equivalent Key

PN Menu: **Marker Function** > **Analysis Range (X)**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:FUNCtion:DOMain:X {FRANge|BDMarker}

:CALCulate:PN[1-1]:TRACe[1-1]:FUNCtion:DOMain:X?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:FUNCtion:DOMain:X FRANge"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:FUNCtion:DOMain:X ?"
30 ENTER 717;A$
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.DOMain.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.DOMain.Y = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.DOMain.Y

Description

This command sets/gets Y analysis range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Y analysis range
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	BDMarker
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.PN1.TRACe1.FUNCtion.DOMain.Y = Var
Var = SCPI.CALCulate.PN1.TRACe1.FUNCtion.DOMain.Y
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.DOMain.X

Equivalent Key

PN Menu: **Marker Function** > **Analysis Range (Y)**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:FUNCtion:DOMain:Y {FRANge|BDMarker}

:CALCulate:PN[1-1]:TRACe[1-1]:FUNCtion:DOMain:Y?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:FUNCtion:DOMain:Y FRANge"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:FUNCtion:DOMain:Y ?"
30 ENTER 717;A$
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.INTegral.DATA_Q integ_noise, freq_range, rms_rad, rms_deg, jitter, residual_fm

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.INTegral.DATA_Q integ_noise, freq_range, rms_rad, rms_deg, jitter, residual_fm

Description

This command gets trace data noise, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*integ_noise*
Description	Integral noise
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No

Error	-

Parameter	*freq_range*
Description	Frequency range
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*rms_rad*
Description	Rms in radians
Data Type	Double precision floating point type (Double)

Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*rms_deg*
Description	RMS in degrees
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No

Error	-

Parameter	*jitter*
Description	Jitter
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*residual_fm*
Description	Residual FM
Data Type	Double precision floating point type (Double)

Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Integ_noise As Double
Dim Freq_range As Double
Dim Rms_rad As Double
Dim Rms_deg As Double
Dim jitter As Double
Dim Residual_fm As Double
SCPI.CALCulate.PN.TRACe.FUNCtion.TYPE = "INTegral"
SCPI.CALCulate.PN.TRACe.FUNCtion.INTegral.DATA_Q Integ_noise, Freq_range, Rms_rad, Rms_deg, jitter,
Residual_fm
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.INTegral.MEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:FUNCtion:INTegral:DATA?

Query Response

{ - - - - - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:FUNCtion:INTegral:DATA ?"
30 ENTER 717;A$
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.INTegral.MEMory_Q integ_noise, freq_range, rms_rad, rms_deg, jitter, residual_fm

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.INTegral.MEMory_Q integ_noise, freq_range, rms_rad, rms_deg, jitter, residual_fm

Description

This command gets memory data noise, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*integ_noise*
Description	Integral noise
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No

Error	-

Parameter	*freq_range*
Description	Frequency range
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*rms_rad*
Description	Rms in radians
Data Type	Double precision floating point type (Double)

Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*rms_deg*
Description	RMS in degrees
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No

Error	-

Parameter	*jitter*
Description	Jitter
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*residual_fm*
Description	Residual FM
Data Type	Double precision floating point type (Double)

Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
Var = SCPI.CALCulate.PN1.TRACe1.FUNCtion.INTegral.MEMory_Q integ_noise, freq_range, rms_rad, rms_deg,
jitter, residual_fm
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.INTegral.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:FUNCtion:INTegral:MEMory?

Query Response

{ - - - - - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:FUNCtion:INTegral:MEMory ?"
30 ENTER 717;A$
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak

Description

This command gets trace data statistics, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*mean*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*std_dev*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*peak_to_peak*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-

Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
Var = SCPI.CALCulate.PN1.TRACe1.FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:FUNCtion:STATistics:DATA?

Query Response

{ - - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:FUNCtion:STATistics:DATA ?"
30 ENTER 717;A$
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak

Description

This command gets memory data statistics, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*mean*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*std_dev*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*peak_to_peak*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-

Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
Var = SCPI.CALCulate.PN1.TRACe1.FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.STATistics.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:FUNCtion:STATistics:MEMory?

Query Response

{ - - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:FUNCtion:STATistics:MEMory ?"
30 ENTER 717;A$
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.TYPE

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.TYPE = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.TYPE

Description

This command sets/gets Analysis type, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Analysis type
Data Type	Character string type (String)
Range	OFF|STATistics|INTegral
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.PN1.TRACe1.FUNCtion.TYPE = Var
Var = SCPI.CALCulate.PN1.TRACe1.FUNCtion.TYPE
```

Equivalent Key

PN Menu: **Marker Function** > **Analysis Type**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:FUNCtion:TYPE {OFF|STATistics|INTegral}

:CALCulate:PN[1-1]:TRACe[1-1]:FUNCtion:TYPE?

Query Response

{OFF|STATistics|INTegral} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:FUNCtion:TYPE OFF"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:FUNCtion:TYPE ?"
30 ENTER 717;A$
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).HOLD

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).HOLD = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).HOLD

Description

This command sets/gets Data hold, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Data hold
Data Type	Character string type (String)
Range	OFF|MAXimum|MINimum
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.PN1.TRACe1.HOLD = Var
Var = SCPI.CALCulate.PN1.TRACe1.HOLD
```

Equivalent Key

PN Menu: **Trace View** > **Data Hold**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:HOLD {OFF|MAXimum|MINimum}

:CALCulate:PN[1-1]:TRACe[1-1]:HOLD?

Query Response

{OFF|MAXimum|MINimum} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:HOLD OFF"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:HOLD ?"
30 ENTER 717;A$
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.FAIL

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.FAIL

Description

This command gets limit test pass/fail, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test result
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Boolean
Var= 1
Var = SCPI.CALCulate.PN1.TRACe1.LIMit.FAIL
```

Related Objects

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:LIMit:FAIL?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:LIMit:FAIL ?"
20 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.LDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.LDATa = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.LDATa

Description

This command sets/gets expanded lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Lower limit line
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.PN1.TRACe1.LIMit.LOWer.LDATa = Var
Var = SCPI.CALCulate.PN1.TRACe1.LIMit.LOWer.LDATa
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:LIMit:LOWer:LDATa {1...1601}

:CALCulate:PN[1-1]:TRACe[1-1]:LIMit:LOWer:LDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PN1:TRACe1:LIMit:LOWer:LDATa 1,1601"
30 OUTPUT 717;":CALCulate:PN1:TRACe1:LIMit:LOWer:LDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

Description

This command clears lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PN1.TRACe1.LIMit.LOWer.SEGMent.CLEar

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:CLEar

Example of use

10 OUTPUT 717;":CALCulate:PN1:TRACe1:LIMit:LOWer:SEGMent:CLEar"

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

Description

This command sets/gets number of lower limit segments, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Lower limit segments
Data Type	Long integer type (Long)
Range	1 ~ 100
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.PN1.TRACe1.LIMit.LOWer.SEGMent.COUNt = Var
Var = SCPI.CALCulate.PN1.TRACe1.LIMit.LOWer.SEGMent.COUNt
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:COUNt {1 ~ 100}

:CALCulate:PN[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:COUNt?

Query Response

{1 ~ 100} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:LIMit:LOWer:SEGMent:COUNt 1 "
20 OUTPUT 717;":CALCulate:PN1:TRACe1:LIMit:LOWer:SEGMent:COUNt ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Description

This command sets/gets lower limit line data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Lower limit data
Data Type	Variant type Array (Range)
Range	1...400
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 400) as Variant
SCPI.CALCulate.PN1.TRACe1.LIMit.LOWer.SEGMent.DATA = Var
Var = SCPI.CALCulate.PN1.TRACe1.LIMit.LOWer.SEGMent.DATA
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:DATA {1...400}

:CALCulate:PN[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:DATA?

Query Response

{1...400} <newline><^END>

Example of use

```
10 Dim A(1:400)
20 OUTPUT 717;":CALCulate:PN1:TRACe1:LIMit:LOWer:SEGMent:DATA 1,400"
30 OUTPUT 717;":CALCulate:PN1:TRACe1:LIMit:LOWer:SEGMent:DATA ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.REPort.DATA

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.REPort.DATA

Description

This command gets pass/fail list, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test report
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
Var = SCPI.CALCulate.PN1.TRACe1.LIMit.REPort.DATA
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:LIMit:REPort[:DATA]?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PN1:TRACe1:LIMit:REPort[:DATA] ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.STATe = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.STATe

Description

This command sets/gets limit test on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.PN1.TRACe1.LIMit.STATe = Var
Var = SCPI.CALCulate.PN1.TRACe1.LIMit.STATe
```

Equivalent Key

PN Menu: **Display** > **Limit Test** > **Limit Test**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:LIMit[:STATe] {ON|OFF|1|0}

:CALCulate:PN[1-1]:TRACe[1-1]:LIMit[:STATe]?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:LIMit[:STATe] 1"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:LIMit[:STATe] ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.UPPer.LDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.UPPer.LDATa = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.UPPer.LDATa

Description

This command sets/gets expanded upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Upper limit line
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.PN1.TRACe1.LIMit.UPPer.LDATa = Var
Var = SCPI.CALCulate.PN1.TRACe1.LIMit.UPPer.LDATa
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:LIMit:UPPer:LDATa {1...1601}

:CALCulate:PN[1-1]:TRACe[1-1]:LIMit:UPPer:LDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PN1:TRACe1:LIMit:UPPer:LDATa 1,1601"
30 OUTPUT 717;":CALCulate:PN1:TRACe1:LIMit:UPPer:LDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

Description

This command clears upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PN1.TRACe1.LIMit.UPPer.SEGMent.CLEar

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:CLEar

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:LIMit:UPPer:SEGMent:CLEar"
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Description

This command sets/gets number of upper limit segments, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Upper limit segment
Data Type	Long integer type (Long)
Range	1 ~ 100
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.PN1.TRACe1.LIMit.UPPer.SEGMent.COUNt = Var
Var = SCPI.CALCulate.PN1.TRACe1.LIMit.UPPer.SEGMent.COUNt
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:COUNt {1 ~ 100}

:CALCulate:PN[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:COUNt?

Query Response

{1 ~ 100} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:LIMit:UPPer:SEGMent:COUNt 1 "
20 OUTPUT 717;":CALCulate:PN1:TRACe1:LIMit:UPPer:SEGMent:COUNt ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Description

This command sets/gets upper limit line data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Upper limit line data
Data Type	Variant type Array (Range)
Range	1...400
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 400) as Variant
SCPI.CALCulate.PN1.TRACe1.LIMit.UPPer.SEGMent.DATA = Var
Var = SCPI.CALCulate.PN1.TRACe1.LIMit.UPPer.SEGMent.DATA
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.PN(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:DATA {1...400}

:CALCulate:PN[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:DATA?

Query Response

{1...400} <newline><^END>

Example of use

```
10 Dim A(1:400)
20 OUTPUT 717;":CALCulate:PN1:TRACe1:LIMit:UPPer:SEGMent:DATA 1,400"
30 OUTPUT 717;":CALCulate:PN1:TRACe1:LIMit:UPPer:SEGMent:DATA ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

Description

This command sets marker search LPEak, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PN1.TRACe1.MARKer10.SEARch.EXECute.LPEak

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

PN Menu: **Marker Search** > **Peak** > **Search Left**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:LPEak

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:SEARch:EXECute:LPEak"
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

Description

This command sets marker search LSPurious, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

This function is allowed to use when the spurious display is power.

Examples

SCPI.CALCulate.PN1.TRACe1.MARKer10.SEARch.EXECute.LSPurious

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

PN Menu: **Marker Search** > **Spurious** > **Search Left**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute.LSPurious

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:SEARch:EXECute.LSPurious"
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

Description

This command sets marker search LTARget, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PN1.TRACe1.MARKer10.SEARch.EXECute.LTARget

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

PN Menu: **Marker Search** > **Target** > **Search Left**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:LTARget

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:SEARch:EXECute:LTARget"
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

Description

This command sets marker search MAXimum, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PN1.TRACe1.MARKer10.SEARch.EXECute.MAXimum

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

PN Menu: **Marker Search** > **Search Max**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:MAXimum

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:SEARch:EXECute:MAXimum"
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

Description

This command sets marker search MINimum, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PN1.TRACe1.MARKer10.SEARch.EXECute.MINimum

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

PN Menu: **Marker Search** > **Search Min**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:MINimum

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:SEARch:EXECute:MINimum"
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

Description

This command sets marker search PEAK, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PN1.TRACe1.MARKer10.SEARch.EXECute.PEAK

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

PN Menu: **Marker Search** > **Peak** > **Search Peak**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:PEAK

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:SEARch:EXECute:PEAK"
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

Description

This command sets marker search RPEak, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PN1.TRACe1.MARKer10.SEARch.EXECute.RPEak

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

PN Menu: **Marker Search** > **Peak** > **Search Right**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:RPEak

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:SEARch:EXECute:RPEak"
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

Description

This command sets marker search RSPurious, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

This function is allowed to use when the spurious display is power.

Examples

SCPI.CALCulate.PN1.TRACe1.MARKer10.SEARch.EXECute.RSPurious

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

PN Menu: **Marker Search** > **Spurious** > **Search Right**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:RSPurious

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:SEARch:EXECute:RSPurious"
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

Description

This command sets marker search RTARget, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PN1.TRACe1.MARKer10.SEARch.EXECute.RTARget

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

PN Menu: **Marker Search** > **Target** > **Search Right**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:RTARget

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:SEARch:EXECute:RTARget"
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

Description

This command sets marker search SPURious, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

NOTE This function is allowed to use when the spurious display is power.

Examples

SCPI.CALCulate.PN1.TRACe1.MARKer10.SEARch.EXECute.SPURious

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

PN Menu: **Marker Search** > **Spurious** > **Search Spurious**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:SPURious

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:SEARch:EXECute:SPURious"
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Description

This command sets marker search TARGet, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PN1.TRACe1.MARKer10.SEARch.EXECute.TARGet

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

Equivalent Key

PN Menu: **Marker Search** > **Target** > **Search Target**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:TARGet

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:SEARch:EXECute:TARGet"
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion

Description

This command sets/gets marker-search-peak excursion value, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search peak excursion value
Data Type	Double precision floating point type (Double)
Range	0 ~ 10G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.PN1.TRACe1.MARKer10.SEARch.PEAK.EXCursion = Var
Var = SCPI.CALCulate.PN1.TRACe1.MARKer10.SEARch.PEAK.EXCursion
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity

Equivalent Key

PN Menu: **Marker Search** > **Peak** > **Peak Excursion**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:PEAK:EXCursion {0 ~ 10G}

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:PEAK:EXCursion?

Query Response

{0 ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:SEARch:PEAK:EXCursion 0"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:SEARch:PEAK:EXCursion ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity

Description

This command sets/gets marker-search-peak polarity type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search peak polarity type
Data Type	Character string type (String)
Range	POSitive|NEGative|BOTH
Preset Value	POSitive
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "POSitive"
SCPI.CALCulate.PN1.TRACe1.MARKer10.SEARch.PEAK.POLarity = Var
Var = SCPI.CALCulate.PN1.TRACe1.MARKer10.SEARch.PEAK.POLarity
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion

Equivalent Key

PN Menu: **Marker Search** > **Peak** > **Peak Polarity**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:PEAK:POLarity {POSitive|NEGative|BOTH}

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:PEAK:POLarity?

Query Response

{POSitive|NEGative|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:SEARch:PEAK:POLarity POSitive"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:SEARch:PEAK:POLarity ?"
30 ENTER 717;A$
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition

Description

This command sets/gets marker-target transition type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker target transition type
Data Type	Character string type (String)
Range	POSitive|NEGative|BOTH
Preset Value	BOTH
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as String
Var= "POSitive"
SCPI.CALCulate.PN1.TRACe1.MARKer10.SEARch.TARGet.TRANsition = Var
Var = SCPI.CALCulate.PN1.TRACe1.MARKer10.SEARch.TARGet.TRANsition
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y

Equivalent Key

PN Menu: **Marker Search** > **Target** > **Target Transition**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TARGet:TRANsition {POSitive|NEGative|BOTH}

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TARGet:TRANsition?

Query Response

{POSitive|NEGative|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:SEARch:TARGet:TRANsition POSitive"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:SEARch:TARGet:TRANsition ?"
30 ENTER 717;A$
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y

Description

This command sets/gets marker target value, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker target value
Data Type	Double precision floating point type (Double)
Range	-10G ~ 10G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.PN1.TRACe1.MARKer10.SEARch.TARGet.Y = Var
Var = SCPI.CALCulate.PN1.TRACe1.MARKer10.SEARch.TARGet.Y
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition

Equivalent Key

PN Menu: **Marker Search** > **Target** > **Target Value**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TARGet:Y {-10G ~ 10G}

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TARGet:Y?

Query Response

{-10G ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:SEARch:TARGet:Y -10000000000"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:SEARch:TARGet:Y ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE

Description

This command sets/gets search tracking type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search tracking type
Data Type	Character string type (String)
Range	OFF|MAXimum|MINimum|PEAK|TARGet|SPURious
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.PN1.TRACe1.MARKer10.SEARch.TRACking.TYPE = Var
Var = SCPI.CALCulate.PN1.TRACe1.MARKer10.SEARch.TRACking.TYPE
```

Related Objects

Equivalent Key

PN Menu: **Marker Search** > **Tracking**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TRACking:TYPE {OFF|MAXimum|MINimum|PEAK|TARGet|SPURious}

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TRACking:TYPE?

Query Response

{OFF|MAXimum|MINimum|PEAK|TARGet|SPURious} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:SEARch:TRACking:TYPE OFF"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:SEARch:TRACking:TYPE ?"
30 ENTER 717;A$
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).STATe = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).STATe

Description

This command sets/gets marker visible on/off, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker visible state
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.PN1.TRACe1.MARKer10.STATe = Var
Var = SCPI.CALCulate.PN1.TRACe1.MARKer10.STATe
```

Equivalent Key

PN Menu: **Marker** > **Clear Marker Menu** > **Marker 1**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:STATe {ON|OFF|1|0}

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:STATe 1"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).X = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).X

Description

This command sets/gets marker x position., for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker X position
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	1k
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
SCPI.CALCulate.PN1.TRACe1.MARKer10.X = Var
Var = SCPI.CALCulate.PN1.TRACe1.MARKer10.X
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).Y

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:X

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:X?

Query Response

{Double} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:X 0"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:X ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).Y

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).Y

Description

This command gets marker y position, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker Y position
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var = SCPI.CALCulate.PN1.TRACe1.MARKer10.Y
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).MARKer(Mk).X

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:MARKer[1-10]:Y?

Query Response

{Double} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:MARKer10:Y ?"
20 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).MATH.FUNCtion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).MATH.FUNCtion = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).MATH.FUNCtion

Description

This command sets/gets math operation type, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Math function operation
Data Type	Character string type (String)
Range	NORMal|SUBTract|DIVide|ADD|MULTiply|USER
Preset Value	NORMal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NORMal"
SCPI.CALCulate.PN1.TRACe1.MATH.FUNCtion = Var
Var = SCPI.CALCulate.PN1.TRACe1.MATH.FUNCtion
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).MATH.MEMorize

SCPI.CALCulate.PN(Ch).TRACe(Tr).MATH.OFFSet

Equivalent Key

PN Menu: **Trace View** > **Data Math**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:MATH:FUNCtion {NORMal|SUBTract|DIVide|ADD|MULTiply|USER}

:CALCulate:PN[1-1]:TRACe[1-1]:MATH:FUNCtion?

Query Response

{NORMal|SUBTract|DIVide|ADD|MULTiply|USER} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:MATH:FUNCtion NORMal"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:MATH:FUNCtion ?"
30 ENTER 717;A$
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).MATH.MEMorize

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).MATH.MEMorize

Description

This command copies data to memory, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PN1.TRACe1.MATH.MEMorize

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).MATH.FUNCtion

SCPI.CALCulate.PN(Ch).TRACe(Tr).MATH.OFFSet

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:MATH:MEMorize

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:MATH:MEMorize"
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).MATH.OFFSet

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).MATH.OFFSet = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).MATH.OFFSet

Description

This command sets/gets Trace Offset, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace Offset
Data Type	Double precision floating point type (Double)
Range	-500G ~ 500G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -500000000000
SCPI.CALCulate.PN1.TRACe1.MATH.OFFSet = Var
Var = SCPI.CALCulate.PN1.TRACe1.MATH.OFFSet
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).MATH.FUNCtion

SCPI.CALCulate.PN(Ch).TRACe(Tr).MATH.MEMorize

Equivalent Key

PN Menu: **Trace View** > **Offset**

PN Menu: **Trace View** > **Marker** > **-Offset**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:MATH:OFFSet {-500G ~ 500G}

:CALCulate:PN[1-1]:TRACe[1-1]:MATH:OFFSet?

Query Response

{-500G ~ 500G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:MATH:OFFSet -500000000000"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:MATH:OFFSet ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).SMOothing.APERture

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).SMOothing.APERture = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).SMOothing.APERture

Description

This command sets/gets Smoothing aperture, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Smoothing aperture
Data Type	Double precision floating point type (Double)
Range	50m ~ 25
Preset Value	1.5
Unit	%
Resolution	10m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0.05
SCPI.CALCulate.PN1.TRACe1.SMOothing.APERture = Var
Var = SCPI.CALCulate.PN1.TRACe1.SMOothing.APERture
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).SMOothing.STATe

Equivalent Key

PN Menu: **Trace View** > **Aperture**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:SMOothing:APERture {50m ~ 25}

:CALCulate:PN[1-1]:TRACe[1-1]:SMOothing:APERture?

Query Response

{50m ~ 25} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:SMOothing:APERture 0.05"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:SMOothing:APERture ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).SMOothing.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).SMOothing.STATe = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).SMOothing.STATe

Description

This command sets/gets Smoothing on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Smoothing (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.PN1.TRACe1.SMOothing.STATe = Var
Var = SCPI.CALCulate.PN1.TRACe1.SMOothing.STATe
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).SMOothing.APERture

Equivalent Key

PN Menu: **Trace View** > **Smoothing**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:SMOothing:STATe {ON|OFF|1|0}

:CALCulate:PN[1-1]:TRACe[1-1]:SMOothing:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:SMOothing:STATe 1"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:SMOothing:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OMISsion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OMISsion = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OMISsion

Description

This command sets/gets Spurious display omission ON/OFF, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious display omission
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.PN1.TRACe1.SPURious.OMISsion = Var
Var = SCPI.CALCulate.PN1.TRACe1.SPURious.OMISsion
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:SPURious:OMISsion {ON|OFF|1|0}

:CALCulate:PN[1-1]:TRACe[1-1]:SPURious:OMISsion?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:SPURious:OMISsion 1"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:SPURious:OMISsion ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

Description

This command clears threshold table, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PN1.TRACe1.SPURious.OSSPur.FLISt.CLEar

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:SPURious:OSSPur:FLISt:CLEar

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:SPURious:OSSPur:FLISt:CLEar"
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

Description

This command sets/gets Spurious Table for Omit Specified Spurious, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious table for omit specified spurious
Data Type	Long integer type (Long)
Range	1 ~ 100
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.PN1.TRACe1.SPURious.OSSPur.FLISt.COUNt = Var
Var = SCPI.CALCulate.PN1.TRACe1.SPURious.OSSPur.FLISt.COUNt
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:SPURious:OSSPur:FLISt:COUNt {1 ~ 100}

:CALCulate:PN[1-1]:TRACe[1-1]:SPURious:OSSPur:FLISt:COUNt?

Query Response

{1 ~ 100} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:SPURious:OSSPur:FLISt:COUNt 1 "
20 OUTPUT 717;":CALCulate:PN1:TRACe1:SPURious:OSSPur:FLISt:COUNt ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

Description

This command sets/gets threshold table, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Threshold table
Data Type	Variant type Array (Range)
Range	1...100
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 100) as Variant
SCPI.CALCulate.PN1.TRACe1.SPURious.OSSPur.FLISt.DATA = Var
Var = SCPI.CALCulate.PN1.TRACe1.SPURious.OSSPur.FLISt.DATA
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:SPURious:OSSPur:FLISt:DATA {1...100}

:CALCulate:PN[1-1]:TRACe[1-1]:SPURious:OSSPur:FLISt:DATA?

Query Response

{1...100} <newline><^END>

Example of use

```
10 Dim A(1:100)
20 OUTPUT 717;":CALCulate:PN1:TRACe1:SPURious:OSSPur:FLISt:DATA 1,100"
30 OUTPUT 717;":CALCulate:PN1:TRACe1:SPURious:OSSPur:FLISt:DATA ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

Description

This command sets/gets Specified Spurious display omission ON/OFF, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious display omission (ON/OFF)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.PN1.TRACe1.SPURious.OSSPur.OMISsion = Var
Var = SCPI.CALCulate.PN1.TRACe1.SPURious.OSSPur.OMISsion
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:SPURious:OSSPur:OMISsion {ON|OFF|1|0}

:CALCulate:PN[1-1]:TRACe[1-1]:SPURious:OSSPur:OMISsion?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:SPURious:OSSPur:OMISsion 1"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:SPURious:OSSPur:OMISsion ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.POWer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.POWer = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.POWer

Description

This command sets/gets Spurious display merging ON/OFF, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious display merging
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.PN1.TRACe1.SPURious.POWer = Var
Var = SCPI.CALCulate.PN1.TRACe1.SPURious.POWer
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:SPURious:POWer {ON|OFF|1|0}

:CALCulate:PN[1-1]:TRACe[1-1]:SPURious:POWer?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:SPURious:POWer 1"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:SPURious:POWer ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.SENSibility

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.SENSibility = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.SENSibility

Description

This command sets/gets spurious sensibility, for the selected trace *Tr* of the selected channel *Ch*.

The low peak has real sensibility when the value is small.

Variable

Parameter	*Value*
Description	Spurious sensibility
Data Type	Double precision floating point type (Double)
Range	10m ~ 10
Preset Value	3
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 5
SCPI.CALCulate.PN1.TRACe1.SPURious.SENSibility = Var
Var = SCPI.CALCulate.PN1.TRACe1.SPURious.SENSibility
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).SMOothing.APERture

SCPI.CALCulate.PN(Ch).TRACe(Tr).SMOothing.STATe

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

PN Menu: **Trace View** > **Spurious** > **Spur Sensibility**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:SPURious:SENSibility {10m ~ 10}

:CALCulate:PN[1-1]:TRACe[1-1]:SPURious:SENSibility?

Query Response

{10m ~ 10} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:SPURious:SENSibility 5"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:SPURious:SENSibility?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.SLISt

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.SLISt

Description

This command gets Spurious List, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious list
Data Type	Variant type Array (Range)
Range	1...0
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 0) as Variant
Var = SCPI.CALCulate.PN1.TRACe1.SPURious.SLISt
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:SPURious:SLISt?

Query Response

{1...0} <newline><^END>

Example of use

```
10 Dim A(1:0)
20 OUTPUT 717;":CALCulate:PN1:TRACe1:SPURious:SLISt ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

Description

This command sets/gets Spurious minimum level definition, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious minimum level definition
Data Type	Double precision floating point type (Double)
Range	-500 ~ 500
Preset Value	-500
Unit	dBc
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.PN1.TRACe1.SPURious.THReshold.LEVel.MINimum = Var
Var = SCPI.CALCulate.PN1.TRACe1.SPURious.THReshold.LEVel.MINimum
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

PN Menu: **Trace View** > **Spurious** > **Minimum Spur Level**

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:SPURious:THReshold:LEVel:MINimum {-500 ~ 500}

:CALCulate:PN[1-1]:TRACe[1-1]:SPURious:THReshold:LEVel:MINimum?

Query Response

{-500 ~ 500} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:SPURious:THReshold:LEVel:MINimum 0"
20 OUTPUT 717;":CALCulate:PN1:TRACe1:SPURious:THReshold:LEVel:MINimum ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

Description

This command clears threshold table, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PN1.TRACe1.SPURious.THReshold.TABLe.CLEar

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:CLEar

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:SPURious:THReshold:TABLe:CLEar"
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

Description

This command sets/gets number of threshold segments, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Number of threshold segments
Data Type	Long integer type (Long)
Range	1 ~ 20
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.PN1.TRACe1.SPURious.THReshold.TABLe.COUNt = Var
Var = SCPI.CALCulate.PN1.TRACe1.SPURious.THReshold.TABLe.COUNt
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:COUNt {1 ~ 20}

:CALCulate:PN[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:COUNt?

Query Response

{1 ~ 20} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PN1:TRACe1:SPURious:THReshold:TABLe:COUNt 1 "
20 OUTPUT 717;":CALCulate:PN1:TRACe1:SPURious:THReshold:TABLe:COUNt ?"
30 ENTER 717;A
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA = *Value*

Value = SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Description

This command sets/gets threshold table, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Threshold table
Data Type	Variant type Array (Range)
Range	1...60
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 60) as Variant
SCPI.CALCulate.PN1.TRACe1.SPURious.THReshold.TABLe.DATA = Var
Var = SCPI.CALCulate.PN1.TRACe1.SPURious.THReshold.TABLe.DATA
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.PN(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PN[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:DATA {1...60}

:CALCulate:PN[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:DATA?

Query Response

{1...60} <newline><^END>

Example of use

```
10 Dim A(1:60)
20 OUTPUT 717;":CALCulate:PN1:TRACe1:SPURious:THReshold:TABLe:DATA 1,60"
30 OUTPUT 717;":CALCulate:PN1:TRACe1:SPURious:THReshold:TABLe:DATA ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PS(Ch).ALLTrace.LIMit.FAIL

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PS(Ch).ALLTrace.LIMit.FAIL

Description

This command gets limit test pass/fail, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Result of limit test
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Boolean
Var = SCPI.CALCulate.PS1.ALLTrace.LIMit.FAIL
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:ALLTrace:LIMit:FAIL?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:ALLTrace:LIMit:FAIL?"
20 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.COUPle.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.COUPle.STATe = *Value*

Value = SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.COUPle.STATe

Description

This command sets/gets marker couple on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker Coupling Function state (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var = 1
SCPI.CALCulate.PS1.ALLTrace.MARKer.COUPle.STATe = Var
Var = SCPI.CALCulate.PS1.ALLTrace.MARKer.COUPle.STATe
```

Related Objects

SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.DISCrete.STATe

SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.REFerence.NUMBer

SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.REFerence.STATe

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:ALLTrace:MARKer:COUPle:STATe {ON|OFF|1|0}

:CALCulate:PS[1-1]:ALLTrace:MARKer:COUPle:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:ALLTrace:MARKer:COUPle:STATe 1"
20 OUTPUT 717;":CALCulate:PS1:ALLTrace:MARKer:COUPle:STATe?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.DISCrete.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.DISCrete.STATe = *Value*

Value = SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.DISCrete.STATe

Description

This command sets/gets marker discrete on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Enables/Disables marker discrete
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var = 1
SCPI.CALCulate.PS1.ALLTrace.MARKer.DISCrete.STATe = Var
Var = SCPI.CALCulate.PS1.ALLTrace.MARKer.DISCrete.STATe
```

Related Objects

SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.COUPle.STATe

SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.REFerence.NUMBer

SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.REFerence.STATe

Equivalent Key

PS Menu: **Marker** > **More Functions** > **Discrete**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:ALLTrace:MARKer:DISCrete:STATe {ON|OFF|1|0}

:CALCulate:PS[1-1]:ALLTrace:MARKer:DISCrete:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:ALLTrace:MARKer:DISCrete:STATe 1"
20 OUTPUT 717;":CALCulate:PS1:ALLTrace:MARKer:DISCrete:STATe?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.REFerence.NUMBer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.REFerence.NUMBer = *Value*

Value = SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.REFerence.NUMBer

Description

This command sets/gets reference marker number., for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker reference number
Data Type	Long integer type (Long)
Range	1 ~ 10
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.PS1.ALLTrace.MARKer.REFerence.NUMBer = Var
Var = SCPI.CALCulate.PS1.ALLTrace.MARKer.REFerence.NUMBer
```

Related Objects

SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.COUPle.STATe

SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.DISCrete.STATe

SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.REFerence.STATe

Equivalent Key

PS Menu: **Marker** > **More Functions** > **Ref Marker**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:ALLTrace:MARKer:REFerence:NUMBer {1 ~ 10}

:CALCulate:PS[1-1]:ALLTrace:MARKer:REFerence:NUMBer?

Query Response

{1 ~ 10} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:ALLTrace:MARKer:REFerence:NUMBer 1 "
20 OUTPUT 717;":CALCulate:PS1:ALLTrace:MARKer:REFerence:NUMBer?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.REFerence.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.REFerence.STATe = *Value*

Value = SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.REFerence.STATe

Description

This command sets/gets delta marker mode on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Delta marker mode (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.PS1.ALLTrace.MARKer.REFerence.STATe = Var
Var = SCPI.CALCulate.PS1.ALLTrace.MARKer.REFerence.STATe
```

Related Objects

SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.COUPle.STATe

SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.DISCrete.STATe

SCPI.CALCulate.PS(Ch).ALLTrace.MARKer.REFerence.NUMBer

Equivalent Key

PS Menu: **Marker** > **More Functions** > **Ref Marker Mode**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:ALLTrace:MARKer:REFerence:STATe {ON|OFF|1|0}

:CALCulate:PS[1-1]:ALLTrace:MARKer:REFerence:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:ALLTrace:MARKer:REFerence:STATe 1"
20 OUTPUT 717;":CALCulate:PS1:ALLTrace:MARKer:REFerence:STATe?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).DATA.CARRier

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).DATA.CARRier = *Value*

Value = SCPI.CALCulate.PS(Ch).DATA.CARRier

Description

This command sets/gets carrier data: freq, power, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Carrier data: Freq & Power
Data Type	Variant type Array (Range)
Range	1...2
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 2) as Variant
SCPI.CALCulate.PS1.DATA.CARRier = Var
Var = SCPI.CALCulate.PS1.DATA.CARRier
```

Related Objects

SCPI.CALCulate.PS(Ch).DATA.PDATa

SCPI.CALCulate.PS(Ch).DATA.RDATa

SCPI.CALCulate.PS(Ch).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:DATA:CARRier {1...2}

:CALCulate:PS[1-1]:DATA:CARRier?

Query Response

{1...2} <newline><^END>

Example of use

```
10 Dim A(1:2)
20 OUTPUT 717;":CALCulate:PS1:DATA:CARRier 1,2"
30 OUTPUT 717;":CALCulate:PS1:DATA:CARRier?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PS(Ch).DATA.PDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).DATA.PDATa = *Value*

Value = SCPI.CALCulate.PS(Ch).DATA.PDATa

Description

This command sets/gets measurement raw power data, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Measurement raw power data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.PS1.DATA.PDATa = Var
Var = SCPI.CALCulate.PS1.DATA.PDATa
```

Related Objects

SCPI.CALCulate.PS(Ch).DATA.CARRier

SCPI.CALCulate.PS(Ch).DATA.RDATa

SCPI.CALCulate.PS(Ch).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:DATA:PDATa {1...1601}

:CALCulate:PS[1-1]:DATA:PDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PS1:DATA:PDATa 1,1601"
30 OUTPUT 717;":CALCulate:PS1:DATA:PDATa?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PS(Ch).DATA.RDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).DATA.RDATa = *Value*

Value = SCPI.CALCulate.PS(Ch).DATA.RDATa

Description

This command sets/gets measurement raw data, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Measurement raw data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.PS1.DATA.RDATa = Var
Var = SCPI.CALCulate.PS1.DATA.RDATa
```

Related Objects

SCPI.CALCulate.PS(Ch).DATA.CARRier

SCPI.CALCulate.PS(Ch).DATA.PDATa

SCPI.CALCulate.PS(Ch).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:DATA:RDATa {1...1601}

:CALCulate:PS[1-1]:DATA:RDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PS1:DATA:RDATa 1,1601"
30 OUTPUT 717;":CALCulate:PS1:DATA:RDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PS(Ch).DATA.XDATa

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PS(Ch).DATA.XDATa

Description

This command gets X axis data, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	X axis data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
Var = SCPI.CALCulate.PS1.DATA.XDATa
```

Related Objects

SCPI.CALCulate.PS(Ch).DATA.CARRier

SCPI.CALCulate.PS(Ch).DATA.PDATa

SCPI.CALCulate.PS(Ch).DATA.RDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:DATA:XDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PS1:DATA:XDATa?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).ALLMarker.ACTive

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).ALLMarker.ACTive = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).ALLMarker.ACTive

Description

This command sets/gets active marker, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Active marker
Data Type	Long integer type (Long)
Range	1 ~ 10
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.PS1.TRACe1.ALLMarker.ACTive = Var
Var = SCPI.CALCulate.PS1.TRACe1.ALLMarker.ACTive
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:ALLMarker:ACTive {1 ~ 10}

:CALCulate:PS[1-1]:TRACe[1-1]:ALLMarker:ACTive?

Query Response

{1 ~ 10} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:ALLMarker:ACTive 1 "
20 OUTPUT 717;":CALCulate:PS1:TRACe1:ALLMarker:ACTive?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

Description

This command sets/gets marker search X range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search X range source
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	FRANge
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.PS1.TRACe1.ALLMarker.SEARch.DOMain.X = Var
Var = SCPI.CALCulate.PS1.TRACe1.ALLMarker.SEARch.DOMain.X
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

SCPI.CALCulate.PS(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Equivalent Key

PS Menu: **Marker Search** > **Search Range (X)**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:ALLMarker:SEARch:DOMain:X {FRANge|BDMarker}

:CALCulate:PS[1-1]:TRACe[1-1]:ALLMarker:SEARch:DOMain:X?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:ALLMarker:SEARch:DOMain:X FRANge"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:ALLMarker:SEARch:DOMain:X?"
30 ENTER 717;A$
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Description

This command sets/gets marker search Y range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search Y range
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	FRANge
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.PS1.TRACe1.ALLMarker.SEARch.DOMain.Y = Var
Var = SCPI.CALCulate.PS1.TRACe1.ALLMarker.SEARch.DOMain.Y
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

SCPI.CALCulate.PS(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Equivalent Key

PS Menu: **Marker Search** > **Search Range (Y)**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:ALLMarker:SEARch:DOMain:Y {FRANge|BDMarker}

:CALCulate:PS[1-1]:TRACe[1-1]:ALLMarker:SEARch:DOMain:Y?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:ALLMarker:SEARch:DOMain:Y FRANge"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:ALLMarker:SEARch:DOMain:Y?"
30 ENTER 717;A$
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Description

This command sets search peak all, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PS1.TRACe1.ALLMarker.SEARch.PEAK

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

SCPI.CALCulate.PS(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Equivalent Key

PS Menu: **Marker Search** > **Peak** > **Search Peak All**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:ALLMarker:SEARch:PEAK

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:ALLMarker:SEARch:PEAK"
```

SCPI.CALCulate.PN(Ch).TRACe(Tr).ALLMarker.SEARch.SPURious

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).ALLMarker.SEARch.SPURious

Description

This command sets search spurious all, for the selected trace *Tr* of the selected channel *Ch*.

You can display up to 10 markers.

NOTE This function is allowed to use when the spurious display is power.

Examples

SCPI.CALCulate.PS1.TRACe1.ALLMarker.SEARch.SPURious

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

Equivalent Key

PS Menu: **Marker Search** > **Spurious** > **Search Spurious All**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:ALLMarker:SEARch:SPURious

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:ALLMarker:SEARch:SPURious"
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.CENTer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.CENTer = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.CENTer

Description

This command sets/gets band marker X center, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X center
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.PS1.TRACe1.BDMarker.X.CENTer = Var
Var = SCPI.CALCulate.PS1.TRACe1.BDMarker.X.CENTer
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

PS Menu: **Marker Search** > **Band Marker X** > **Center**

PS Menu: **Marker Function** > **Band Marker X** > **Center**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:BDMarker:X:CENTer {-1T ~ 1T}

:CALCulate:PS[1-1]:TRACe[1-1]:BDMarker:X:CENTer?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:BDMarker:X:CENTer -10000000000"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:BDMarker:X:CENTer?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.SPAN

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.SPAN = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.SPAN

Description

This command sets/gets band marker X span, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X span
Data Type	Double precision floating point type (Double)
Range	0 ~ 2T
Preset Value	2T
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.PS1.TRACe1.BDMarker.X.SPAN = Var
Var = SCPI.CALCulate.PS1.TRACe1.BDMarker.X.SPAN
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

PS Menu: **Marker Search** > **Band Marker X** > **Span**

PS Menu: **Marker Function** > **Band Marker X** > **Span**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:BDMarker:X:SPAN {0 ~ 2T}

:CALCulate:PS[1-1]:TRACe[1-1]:BDMarker:X:SPAN?

Query Response

{0 ~ 2T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:BDMarker:X:SPAN 0"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:BDMarker:X:SPAN?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.STARt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.STARt = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.STARt

Description

This command sets/gets band marker X start, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X start
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	-1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.PS1.TRACe1.BDMarker.X.STARt = Var
Var = SCPI.CALCulate.PS1.TRACe1.BDMarker.X.STARt
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

PS Menu: **Marker Search** > **Band Marker X** > **Start**

PS Menu: **Marker Function** > **Band Marker X** > **Start**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:BDMarker:X:STARt {-1T ~ 1T}

:CALCulate:PS[1-1]:TRACe[1-1]:BDMarker:X:STARt?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:BDMarker:X:STARt -10000000000"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:BDMarker:X:STARt?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.STATe = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.STATe

Description

This command sets/gets band marker visible on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker visibility (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.PS1.TRACe1.BDMarker.X.STATe = Var
Var = SCPI.CALCulate.PS1.TRACe1.BDMarker.X.STATe
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

PS Menu: **Marker Search** > **Band Marker X** > **Band Marker X**

PS Menu: **Marker Function** > **Band Marker X** > **Band Marker X**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:BDMarker:X:STATe {ON|OFF|1|0}

:CALCulate:PS[1-1]:TRACe[1-1]:BDMarker:X:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:BDMarker:X:STATe 1"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:BDMarker:X:STATe?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.STOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.STOP = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.STOP

Description

This command sets/gets band marker X stop, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X stop
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.PS1.TRACe1.BDMarker.X.STOP = Var
Var = SCPI.CALCulate.PS1.TRACe1.BDMarker.X.STOP
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.X.STATe

Equivalent Key

PS Menu: **Marker Search** > **Band Marker X** > **Stop**

PS Menu: **Marker Function** > **Band Marker X** > **Stop**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:BDMarker:X:STOP {-1T ~ 1T}

:CALCulate:PS[1-1]:TRACe[1-1]:BDMarker:X:STOP?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:BDMarker:X:STOP -10000000000"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:BDMarker:X:STOP?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.CENTer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.CENTer = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.CENTer

Description

This command sets/gets band marker Y center, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y center
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.PS1.TRACe1.BDMarker.Y.CENTer = Var
Var = SCPI.CALCulate.PS1.TRACe1.BDMarker.Y.CENTer
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

PS Menu: **Marker Search** > **Band Marker Y** > **Center**

PS Menu: **Marker Function** > **Band Marker Y** > **Center**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:BDMarker:Y:CENTer {-1T ~ 1T}

:CALCulate:PS[1-1]:TRACe[1-1]:BDMarker:Y:CENTer?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:BDMarker:Y:CENTer -10000000000"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:BDMarker:Y:CENTer?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.SPAN

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.SPAN = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.SPAN

Description

This command sets/gets band marker Y span, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y span
Data Type	Double precision floating point type (Double)
Range	0 ~ 2T
Preset Value	2T
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.PS1.TRACe1.BDMarker.Y.SPAN = Var
Var = SCPI.CALCulate.PS1.TRACe1.BDMarker.Y.SPAN
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

PS Menu: **Marker Search** > **Band Marker Y** > **Span**

PS Menu: **Marker Function** > **Band Marker Y** > **Span**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:BDMarker:Y:SPAN {0 ~ 2T}

:CALCulate:PS[1-1]:TRACe[1-1]:BDMarker:Y:SPAN?

Query Response

{0 ~ 2T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:BDMarker:Y:SPAN 0"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:BDMarker:Y:SPAN?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.STARt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.STARt = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.STARt

Description

This command sets/gets band marker Y start, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y start
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	-1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.PS1.TRACe1.BDMarker.Y.STARt = Var
Var = SCPI.CALCulate.PS1.TRACe1.BDMarker.Y.STARt
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

PS Menu: **Marker Search** > **Band Marker Y** > **Start**

PS Menu: **Marker Function** > **Band Marker Y** > **Start**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:BDMarker:Y:STARt {-1T ~ 1T}

:CALCulate:PS[1-1]:TRACe[1-1]:BDMarker:Y:STARt?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:BDMarker:Y:STARt -10000000000"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:BDMarker:Y:STARt?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.STATe = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.STATe

Description

This command sets/gets band marker visible on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y state
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.PS1.TRACe1.BDMarker.Y.STATe = Var
Var = SCPI.CALCulate.PS1.TRACe1.BDMarker.Y.STATe
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

PS Menu: **Marker Search** > **Band Marker Y** > **Band Marker Y**

PS Menu: **Marker Function** > **Band Marker Y** > **Band Marker Y**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:BDMarker:Y:STATe {ON|OFF|1|0}

:CALCulate:PS[1-1]:TRACe[1-1]:BDMarker:Y:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:BDMarker:Y:STATe 1"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:BDMarker:Y:STATe?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.STOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.STOP = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.STOP

Description

This command sets/gets band marker Y stop, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y stop
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.PS1.TRACe1.BDMarker.Y.STOP = Var
Var = SCPI.CALCulate.PS1.TRACe1.BDMarker.Y.STOP
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.PS(Ch).TRACe(Tr).BDMarker.Y.STATe

Equivalent Key

PS Menu: **Marker Search** > **Band Marker Y** > **Stop**

PS Menu: **Marker Function** > **Band Marker Y** > **Stop**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:BDMarker:Y:STOP {-1T ~ 1T}

:CALCulate:PS[1-1]:TRACe[1-1]:BDMarker:Y:STOP?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:BDMarker:Y:STOP -10000000000"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:BDMarker:Y:STOP?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.COPY

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.COPY = *Value*

Description

This command copied data to user trace, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Copies data to user trace
Data Type	Long integer type (Long)
Range	1 ~ 8
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.PS1.TRACe1.DATA.COPY = Var
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:DATA:COPY {1 ~ 8}

Query Response

{1 ~ 8} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:DATA:COPY 1"
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.FDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.FDATa = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.FDATa

Description

This command sets/gets formatted trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Formatted trace data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.PS1.TRACe1.DATA.FDATa = Var
Var = SCPI.CALCulate.PS1.TRACe1.DATA.FDATa
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:DATA:FDATa {1...1601}

:CALCulate:PS[1-1]:TRACe[1-1]:DATA:FDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PS1:TRACe1:DATA:FDATa 1,1601"
30 OUTPUT 717;":CALCulate:PS1:TRACe1:DATA:FDATa?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.FMEMory

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.FMEMory = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.FMEMory

Description

This command sets/gets formatted memory data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Formatted memory data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.PS1.TRACe1.DATA.FMEMory = Var
Var = SCPI.CALCulate.PS1.TRACe1.DATA.FMEMory
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:DATA:FMEMory {1...1601}

:CALCulate:PS[1-1]:TRACe[1-1]:DATA:FMEMory?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PS1:TRACe1:DATA:FMEMory 1,1601"
30 OUTPUT 717;":CALCulate:PS1:TRACe1:DATA:FMEMory?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.PDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.PDATa = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.PDATa

Description

This command sets/gets dBc data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	dBC data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.PS1.TRACe1.DATA.PDATa = Var
Var = SCPI.CALCulate.PS1.TRACe1.DATA.PDATa
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:DATA:PDATa {1...1601}

:CALCulate:PS[1-1]:TRACe[1-1]:DATA:PDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PS1:TRACe1:DATA:PDATa 1,1601"
30 OUTPUT 717;":CALCulate:PS1:TRACe1:DATA:PDATa?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.PMEMory

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.PMEMory = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.PMEMory

Description

This command sets/gets dBc memory data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	dBc memory data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.PS1.TRACe1.DATA.PMEMory = Var
Var = SCPI.CALCulate.PS1.TRACe1.DATA.PMEMory
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:DATA:PMEMory {1...1601}

:CALCulate:PS[1-1]:TRACe[1-1]:DATA:PMEMory?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PS1:TRACe1:DATA:PMEMory 1,1601"
30 OUTPUT 717;":CALCulate:PS1:TRACe1:DATA:PMEMory ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.SDATa

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.SDATa

Description

This command gets spurious data (0 or 1), for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
Var = SCPI.CALCulate.PS1.TRACe1.DATA.SDATa
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:DATA:SDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PS1:TRACe1:DATA:SDATa?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.SMEMory

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.SMEMory

Description

This command gets spurious memory data (0 or 1), for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious memory data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
Var = SCPI.CALCulate.PS1.TRACe1.DATA.SMEMory
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:DATA:SMEMory?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PS1:TRACe1:DATA:SMEMory?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.UDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.UDATa = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.UDATa

Description

This command sets/gets unformatted trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Unformatted trace data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.PS1.TRACe1.DATA.UDATa = Var
Var = SCPI.CALCulate.PS1.TRACe1.DATA.UDATa
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:DATA:UDATa {1...1601}

:CALCulate:PS[1-1]:TRACe[1-1]:DATA:UDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PS1:TRACe1:DATA:UDATa 1,1601"
30 OUTPUT 717;":CALCulate:PS1:TRACe1:DATA:UDATa?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.UMEMory

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.UMEMory = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.UMEMory

Description

This command sets/gets unformatted memory data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Unformatted memory data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.PS1.TRACe1.DATA.UMEMory = Var
Var = SCPI.CALCulate.PS1.TRACe1.DATA.UMEMory
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.PDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.PMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.SDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.SMEMory

SCPI.CALCulate.PS(Ch).TRACe(Tr).DATA.UDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:DATA:UMEMory {1...1601}

:CALCulate:PS[1-1]:TRACe[1-1]:DATA:UMEMory?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PS1:TRACe1:DATA:UMEMory 1,1601"
30 OUTPUT 717;":CALCulate:PS1:TRACe1:DATA:UMEMory?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).EQUation.TEXT

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).EQUation.TEXT = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).EQUation.TEXT

Description

This command sets/gets equation string, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Equation string
Data Type	Character string type (String)
Range	254 chars
Preset Value	""
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	170, "Invalid equation label" 171, "Invalid equation

	expression"

Examples

```
Dim Var as String
Var = "MyTr = DATA-MEM"
SCPI.CALCulate.PS1.TRACe1.EQUation.TEXT = Var
Var = SCPI.CALCulate.PS1.TRACe1.EQUation.TEXT
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:EQUation:TEXT

:CALCulate:PS[1-1]:TRACe[1-1]:EQUation:TEXT?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:EQUation:TEXT Sample_Text"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:EQUation:TEXT?"
30 ENTER 717;A$
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.AVARiance.DATA_Q avg_time, fcutoff, allan_var, jitter

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.AVARiance.DATA_Q avg_time, fcutoff, allan_var, jitter

Description

This command gets Allan Variance (data trace), for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*avg_time*
Description	Average time
Data Type	Double precision floating point type (Double)
Range	1n ~ 1k
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*fcutoff*
Description	Frequency cutoff
Data Type	Double precision floating point type (Double)
Range	0 ~ 1G
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*allan_var*
Description	Allan variance
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-

Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*jitter*
Description	Jitter
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim avg_time As Double
Dim fcutoff As Double
Dim allan_var As Double
Dim jitter As Double
```

```
Dim var As Variant
avg_time = 0.001 'Tau = 0.001 sec
fcutoff = 10000 '10 kHz cutoff frequency
SCPI.CALCulate.PS.TRACe.FUNCtion.AVARiance.DATA_Q avg_time, fcutoff, allan_var, jitter
MsgBox "Allan Variance = " & Str$(allan_var) & " " & "Jitter(rms) = " & Str$(jitter) & " (sec)"
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.AVARiance.MEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:FUNCtion:AVARiance:DATA?

Query Response

{1n ~ 1k 0 ~ 1G - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:FUNCtion:AVARiance:DATA?"
30 ENTER 717;A$
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.AVARiance.MEMory_Q avg_time, fcutoff, allan_var, jitter

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.AVARiance.MEMory_Q avg_time, fcutoff, allan_var, jitter

Description

This command gets Allan Variance (memory trace), for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*avg_time*
Description	Average time
Data Type	Double precision floating point type (Double)
Range	1n ~ 1k
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*fcutoff*
Description	Frequency cutoff
Data Type	Double precision floating point type (Double)
Range	0 ~ 1G
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*allan_var*
Description	Allan variance
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-

Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*jitter*
Description	Jitter
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
Var = SCPI.CALCulate.PS1.TRACe1.FUNCtion.AVARiance.MEMory_Q avg_time, fcutoff, allan_var, jitter
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.AVARiance.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:FUNCtion:AVARiance:MEMory?

Query Response

{1n ~ 1k 0 ~ 1G - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:FUNCtion:AVARiance:MEMory?"
30 ENTER 717;A$
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.DOMain.X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.DOMain.X = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.DOMain.X

Description

This command sets/gets X analysis range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	X analysis range
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	BDMarker
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.PS1.TRACe1.FUNCtion.DOMain.X = Var
Var = SCPI.CALCulate.PS1.TRACe1.FUNCtion.DOMain.X
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.DOMain.Y

Equivalent Key

PS Menu: **Marker Function** > **Analysis Range (X)**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:FUNCtion:DOMain:X {FRANge|BDMarker}

:CALCulate:PS[1-1]:TRACe[1-1]:FUNCtion:DOMain:X?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:FUNCtion:DOMain:X FRANge"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:FUNCtion:DOMain:X?"
30 ENTER 717;A$
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.DOMain.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.DOMain.Y = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.DOMain.Y

Description

This command sets/gets Y analysis range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Y analysis range
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	BDMarker
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.PS1.TRACe1.FUNCtion.DOMain.Y = Var
Var = SCPI.CALCulate.PS1.TRACe1.FUNCtion.DOMain.Y
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.DOMain.X

Equivalent Key

PS Menu: **Marker Function** > **Analysis Range (Y)**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:FUNCtion:DOMain:Y {FRANge|BDMarker}

:CALCulate:PS[1-1]:TRACe[1-1]:FUNCtion:DOMain:Y?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:FUNCtion:DOMain:Y FRANge"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:FUNCtion:DOMain:Y?"
30 ENTER 717;A$
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.INTegral.DATA_Q integ_noise, freq_range, rms_rad, rms_deg, jitter, residual_fm

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.INTegral.DATA_Q integ_noise, freq_range, rms_rad, rms_deg, jitter, residual_fm

Description

This command gets trace data noise, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*integ_noise*
Description	Integral noise
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No

Error	-

Parameter	*freq_range*
Description	Frequency range
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*rms-rad*
Description	Rms in radians
Data Type	Double precision floating point type (Double)

Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*rms_deg*
Description	RMS in degrees
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No

Error	-

Parameter	*jitter*
Description	Jitter
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*residual_fm*
Description	Residual FM
Data Type	Double precision floating point type (Double)

Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Integ_noise As Double
Dim Freq_range As Double
Dim Rms_rad As Double
Dim Rms_deg As Double
Dim jitter As Double
Dim Residual_fm As Double
SCPI.CALCulate.PS.TRACe.FUNCtion.TYPE = "INTegral"
SCPI.CALCulate.PS.TRACe.FUNCtion.INTegral.DATA_Q Integ_noise, Freq_range, Rms_rad, Rms_deg, jitter,
Residual_fm
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.INTegral.MEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:FUNCtion:INTegral:DATA?

Query Response

{ - - - - - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:FUNCtion:INTegral:DATA ?"
30 ENTER 717;A$
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.INTegral.MEMory_Q integ_noise, freq_range, rms_rad, rms_deg, jitter, residual_fm

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.INTegral.MEMory_Q integ_noise, freq_range, rms_rad, rms_deg, jitter, residual_fm

Description

This command gets memory data noise, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*integ_noise*
Description	Integral noise
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No

Error	-

Parameter	*freq_range*
Description	Frequency range
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*rms_rad*
Description	Rms in radians
Data Type	Double precision floating point type (Double)

Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*rms_deg*
Description	RMS in degrees
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No

Error	-

Parameter	*jitter*
Description	Jitter
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*residual_fm*
Description	Residual FM
Data Type	Double precision floating point type (Double)

Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
Var = SCPI.CALCulate.PS1.TRACe1.FUNCtion.INTegral.MEMory_Q integ_noise, freq_range, rms_rad, rms_deg,
jitter, residual_fm
```

Related Objects

SCPI.CALCulate.PN(Ch).TRACe(Tr).FUNCtion.INTegral.DATA

SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.INTegral.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:FUNCtion:INTegral:MEMory?

Query Response

{ - - - - - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:FUNCtion:INTegral:MEMory?"
30 ENTER 717;A$
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak

Description

This command gets trace data statistics, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*mean*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No

Error	-

Parameter	*std_dev*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*peak_to_peak*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-

Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
Var = SCPI.CALCulate.PS1.TRACe1.FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:FUNCtion:STATistics:DATA?

Query Response

{ - - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:FUNCtion:STATistics:DATA ?"
30 ENTER 717;A$
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak

Description

This command gets memory data statistics, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*mean*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*std_dev*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*peak_to_peak*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-

Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
Var = SCPI.CALCulate.PS1.TRACe1.FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.STATistics.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:FUNCtion:STATistics:MEMory?

Query Response

{ - - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:FUNCtion:STATistics:MEMory?"
30 ENTER 717;A$
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.TYPE

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.TYPE = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).FUNCtion.TYPE

Description

This command sets/gets Analysis type, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Analysis type
Data Type	Character string type (String)
Range	OFF|STATistics|INTegral
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.PS1.TRACe1.FUNCtion.TYPE = Var
Var = SCPI.CALCulate.PS1.TRACe1.FUNCtion.TYPE
```

Equivalent Key

PS Menu: **Marker Function** > **Analysis Type**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:FUNCtion:TYPE {OFF|STATistics|INTegral}

:CALCulate:PS[1-1]:TRACe[1-1]:FUNCtion:TYPE?

Query Response

{OFF|STATistics|INTegral} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:FUNCtion:TYPE OFF"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:FUNCtion:TYPE ?"
30 ENTER 717;A$
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).HOLD

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).HOLD = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).HOLD

Description

This command sets/gets Data hold, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Data hold
Data Type	Character string type (String)
Range	OFF|MAXimum|MINimum
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.PS1.TRACe1.HOLD = Var
Var = SCPI.CALCulate.PS1.TRACe1.HOLD
```

Equivalent Key

PN Menu: **Trace View** > **Data Hold**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:HOLD {OFF|MAXimum|MINimum}

:CALCulate:PS[1-1]:TRACe[1-1]:HOLD?

Query Response

{OFF|MAXimum|MINimum} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:HOLD OFF"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:HOLD ?"
30 ENTER 717;A$
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.FAIL

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.FAIL

Description

This command gets limit test pass/fail, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test result
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Boolean
Var= 1
Var = SCPI.CALCulate.PS1.TRACe1.LIMit.FAIL
```

Related Objects

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:LIMit:FAIL?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:LIMit:FAIL ?"
20 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.LOWer.LDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.LOWer.LDATa = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.LOWer.LDATa

Description

This command sets/gets expanded lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Lower limit line
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.PS1.TRACe1.LIMit.LOWer.LDATa = Var
Var = SCPI.CALCulate.PS1.TRACe1.LIMit.LOWer.LDATa
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:LIMit:LOWer:LDATa {1...1601}

:CALCulate:PS[1-1]:TRACe[1-1]:LIMit:LOWer:LDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PS1:TRACe1:LIMit:LOWer:LDATa 1,1601"
30 OUTPUT 717;":CALCulate:PS1:TRACe1:LIMit:LOWer:LDATa?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

Description

This command clears lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PS1.TRACe1.LIMit.LOWer.SEGMent.CLEar

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:CLEar

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:LIMit:LOWer:SEGMent:CLEar"
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

Description

This command sets/gets number of lower limit segments, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Lower limit segments
Data Type	Long integer type (Long)
Range	1 ~ 100
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.PS1.TRACe1.LIMit.LOWer.SEGMent.COUNt = Var
Var = SCPI.CALCulate.PS1.TRACe1.LIMit.LOWer.SEGMent.COUNt
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:COUNt {1 ~ 100}

:CALCulate:PS[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:COUNt?

Query Response

{1 ~ 100} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:LIMit:LOWer:SEGMent:COUNt 1 "
20 OUTPUT 717;":CALCulate:PS1:TRACe1:LIMit:LOWer:SEGMent:COUNt?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Description

This command sets/gets lower limit line data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Lower limit data
Data Type	Variant type Array (Range)
Range	1...400
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 400) as Variant
SCPI.CALCulate.PS1.TRACe1.LIMit.LOWer.SEGMent.DATA = Var
Var = SCPI.CALCulate.PS1.TRACe1.LIMit.LOWer.SEGMent.DATA
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:DATA {1...400}

:CALCulate:PS[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:DATA?

Query Response

{1...400} <newline><^END>

Example of use

```
10 Dim A(1:400)
20 OUTPUT 717;":CALCulate:PS1:TRACe1:LIMit:LOWer:SEGMent:DATA 1,400"
30 OUTPUT 717;":CALCulate:PS1:TRACe1:LIMit:LOWer:SEGMent:DATA?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.REPort.DATA

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.REPort.DATA

Description

This command gets pass/fail list, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test report
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
Var = SCPI.CALCulate.PS1.TRACe1.LIMit.REPort.DATA
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:LIMit:REPort[:DATA]?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PS1:TRACe1:LIMit:REPort[:DATA] ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.STATe = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.STATe

Description

This command sets/gets limit test on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.PS1.TRACe1.LIMit.STATe = Var
Var = SCPI.CALCulate.PS1.TRACe1.LIMit.STATe
```

Equivalent Key

PS Menu: **Display** > **Limit Test** > **Limit Test**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:LIMit[:STATe] {ON|OFF|1|0}

:CALCulate:PS[1-1]:TRACe[1-1]:LIMit[:STATe]?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:LIMit[:STATe] 1"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:LIMit[:STATe]?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.UPPer.LDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.UPPer.LDATa = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.UPPer.LDATa

Description

This command sets/gets expanded upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Upper limit line
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.PS1.TRACe1.LIMit.UPPer.LDATa = Var
Var = SCPI.CALCulate.PS1.TRACe1.LIMit.UPPer.LDATa
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:LIMit:UPPer:LDATa {1...1601}

:CALCulate:PS[1-1]:TRACe[1-1]:LIMit:UPPer:LDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:PS1:TRACe1:LIMit:UPPer:LDATa 1,1601"
30 OUTPUT 717;":CALCulate:PS1:TRACe1:LIMit:UPPer:LDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

Description

This command clears upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PS1.TRACe1.LIMit.UPPer.SEGMent.CLEar

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:CLEar

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:LIMit:UPPer:SEGMent:CLEar"
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Description

This command sets/gets number of upper limit segments, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Upper limit segment
Data Type	Long integer type (Long)
Range	1 ~ 100
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.PS1.TRACe1.LIMit.UPPer.SEGMent.COUNt = Var
Var = SCPI.CALCulate.PS1.TRACe1.LIMit.UPPer.SEGMent.COUNt
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:COUNt {1 ~ 100}

:CALCulate:PS[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:COUNt?

Query Response

{1 ~ 100} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:LIMit:UPPer:SEGMent:COUNt 1 "
20 OUTPUT 717;":CALCulate:PS1:TRACe1:LIMit:UPPer:SEGMent:COUNt?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Description

This command sets/gets upper limit line data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Upper limit line data
Data Type	Variant type Array (Range)
Range	1...400
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 400) as Variant
SCPI.CALCulate.PS1.TRACe1.LIMit.UPPer.SEGMent.DATA = Var
Var = SCPI.CALCulate.PS1.TRACe1.LIMit.UPPer.SEGMent.DATA
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.PS(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:DATA {1...400}

:CALCulate:PS[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:DATA?

Query Response

{1...400} <newline><^END>

Example of use

```
10 Dim A(1:400)
20 OUTPUT 717;":CALCulate:PS1:TRACe1:LIMit:UPPer:SEGMent:DATA 1,400"
30 OUTPUT 717;":CALCulate:PS1:TRACe1:LIMit:UPPer:SEGMent:DATA?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

Description

This command sets marker search LPEak, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PS1.TRACe1.MARKer10.SEARch.EXECute.LPEak

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

PS Menu: **Marker Search** > **Peak** > **Search Left**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:LPEak

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:SEARch:EXECute:LPEak"
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

Description

This command sets marker search LSPurious, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

NOTE This function is allowed to use when the spurious display is power.

Examples

SCPI.CALCulate.PS1.TRACe1.MARKer10.SEARch.EXECute.LSPurious

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

PS Menu: **Marker Search** > **Spurious** > **Search Left**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute.LSPurious

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:SEARch:EXECute.LSPurious"
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

Description

This command sets marker search LTARget, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PS1.TRACe1.MARKer10.SEARch.EXECute.LTARget

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

PS Menu: **Marker Search** > **Target** > **Search Left**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:LTARget

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:SEARch:EXECute:LTARget"
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

Description

This command sets marker search MAXimum, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PS1.TRACe1.MARKer10.SEARch.EXECute.MAXimum

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

PS Menu: **Marker Search** > **Search Max**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:MAXimum

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:SEARch:EXECute:MAXimum"
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

Description

This command sets marker search MINimum, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PS1.TRACe1.MARKer10.SEARch.EXECute.MINimum

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

PS Menu: **Marker Search** > **Search Min**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:MINimum

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:SEARch:EXECute:MINimum"
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

Description

This command sets marker search PEAK, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PS1.TRACe1.MARKer10.SEARch.EXECute.PEAK

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

PS Menu: **Marker Search** > **Peak** > **Search Peak**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:PEAK

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:SEARch:EXECute:PEAK"
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

Description

This command sets marker search RPEak, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PS1.TRACe1.MARKer10.SEARch.EXECute.RPEak

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

PS Menu: **Marker Search** > **Peak** > **Search Right**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:RPEak

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:SEARch:EXECute:RPEak"
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

Description

This command sets marker search RSPurious, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

NOTE This function is allowed to use when the spurious display is power.

Examples

SCPI.CALCulate.PS1.TRACe1.MARKer10.SEARch.EXECute.RSPurious

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

PS Menu: **Marker Search** > **Spurious** > **Search Right**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:RSPurious

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:SEARch:EXECute:RSPurious"
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

Description

This command sets marker search RTARget, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PS1.TRACe1.MARKer10.SEARch.EXECute.RTARget

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

PS Menu: **Marker Search** > **Target** > **Search Right**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:RTARget

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:SEARch:EXECute:RTARget"
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

Description

This command sets marker search SPURious, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

NOTE This function is allowed to use when the spurious display is power.

Examples

SCPI.CALCulate.PS1.TRACe1.MARKer10.SEARch.EXECute.SPURious

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

PS Menu: **Marker Search** > **Spurious** > **Search Spurious**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:SPURious

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:SEARch:EXECute:SPURious"
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Description

This command sets marker search TARGet, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PS1.TRACe1.MARKer10.SEARch.EXECute.TARGet

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LSPurious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RSPurious

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.SPURious

Equivalent Key

PS Menu: **Marker Search** > **Target** > **Search Target**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:TARGet

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:SEARch:EXECute:TARGet"
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion

Description

This command sets/gets marker-search-peak excursion value, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search peak excursion value
Data Type	Double precision floating point type (Double)
Range	0 ~ 10G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.PS1.TRACe1.MARKer10.SEARch.PEAK.EXCursion = Var
Var = SCPI.CALCulate.PS1.TRACe1.MARKer10.SEARch.PEAK.EXCursion
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity

Equivalent Key

PS Menu: **Marker Search** > **Peak** > **Peak Excursion**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:PEAK:EXCursion {0 ~ 10G}

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:PEAK:EXCursion?

Query Response

{0 ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:SEARch:PEAK:EXCursion 0"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:SEARch:PEAK:EXCursion ?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity

Description

This command sets/gets marker-search-peak polarity type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search peak polarity type
Data Type	Character string type (String)
Range	POSitive|NEGative|BOTH
Preset Value	POSitive
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "POSitive"
SCPI.CALCulate.PS1.TRACe1.MARKer10.SEARch.PEAK.POLarity = Var
Var = SCPI.CALCulate.PS1.TRACe1.MARKer10.SEARch.PEAK.POLarity
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion

Equivalent Key

PS Menu: **Marker Search** > **Peak** > **Peak Polarity**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:PEAK:POLarity {POSitive|NEGative|BOTH}

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:PEAK:POLarity?

Query Response

{POSitive|NEGative|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:SEARch:PEAK:POLarity POSitive"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:SEARch:PEAK:POLarity ?"
30 ENTER 717;A$
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition

Description

This command sets/gets marker-target transition type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker target transition type
Data Type	Character string type (String)
Range	POSitive|NEGative|BOTH
Preset Value	BOTH
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as String
Var= "POSitive"
SCPI.CALCulate.PS1.TRACe1.MARKer10.SEARch.TARGet.TRANsition = Var
Var = SCPI.CALCulate.PS1.TRACe1.MARKer10.SEARch.TARGet.TRANsition
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y

Equivalent Key

PS Menu: **Marker Search** > **Target** > **Target Transition**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TARGet:TRANsition {POSitive|NEGative|BOTH}

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TARGet:TRANsition?

Query Response

{POSitive|NEGative|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:SEARch:TARGet:TRANsition POSitive"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:SEARch:TARGet:TRANsition ?"
30 ENTER 717;A$
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y

Description

This command sets/gets marker target value, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker target value
Data Type	Double precision floating point type (Double)
Range	-10G ~ 10G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.PS1.TRACe1.MARKer10.SEARch.TARGet.Y = Var
Var = SCPI.CALCulate.PS1.TRACe1.MARKer10.SEARch.TARGet.Y
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition

Equivalent Key

PS Menu: **Marker Search** > **Target** > **Target Value**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TARGet:Y {-10G ~ 10G}

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TARGet:Y?

Query Response

{-10G ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:SEARch:TARGet:Y -10000000000"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:SEARch:TARGet:Y?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE

Description

This command sets/gets search tracking type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search tracking type
Data Type	Character string type (String)
Range	OFF|MAXimum|MINimum|PEAK|TARGet|SPURious
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.PS1.TRACe1.MARKer10.SEARch.TRACking.TYPE = Var
Var = SCPI.CALCulate.PS1.TRACe1.MARKer10.SEARch.TRACking.TYPE
```

Related Objects

Equivalent Key

PS Menu: **Marker Search** > **Tracking**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TRACking:TYPE {OFF|MAXimum|MINimum|PEAK|TARGet|SPURious}

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TRACking:TYPE?

Query Response

{OFF|MAXimum|MINimum|PEAK|TARGet|SPURious} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:SEARch:TRACking:TYPE OFF"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:SEARch:TRACking:TYPE?"
30 ENTER 717;A$
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).STATe = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).STATe

Description

This command sets/gets marker visible on/off, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker visible state
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.PS1.TRACe1.MARKer10.STATe = Var
Var = SCPI.CALCulate.PS1.TRACe1.MARKer10.STATe
```

Equivalent Key

PS Menu: **Marker** > **Clear Marker Menu** > **Marker 1**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:STATe {ON|OFF|1|0}

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:STATe 1"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:STATe?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).X = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).X

Description

This command sets/gets marker x position., for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker X position
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	1k
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var = 100
SCPI.CALCulate.PS1.TRACe1.MARKer10.X = Var
Var = SCPI.CALCulate.PS1.TRACe1.MARKer10.X
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).Y

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:X

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:X?

Query Response

{Double} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:X 0"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:X?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).Y

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).Y

Description

This command gets marker y position, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker Y position
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var = SCPI.CALCulate.PS1.TRACe1.MARKer10.Y
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).MARKer(Mk).X

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:MARKer[1-10]:Y?

Query Response

{Double} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:MARKer10:Y ?"
20 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).MATH.FUNCtion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).MATH.FUNCtion = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).MATH.FUNCtion

Description

This command sets/gets math operation type, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Math function operation
Data Type	Character string type (String)
Range	NORMal|SUBTract|DIVide|ADD|MULTiply|USER
Preset Value	NORMal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NORMal"
SCPI.CALCulate.PS1.TRACe1.MATH.FUNCtion = Var
Var = SCPI.CALCulate.PS1.TRACe1.MATH.FUNCtion
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).MATH.MEMorize

SCPI.CALCulate.PS(Ch).TRACe(Tr).MATH.OFFSet

Equivalent Key

PS Menu: **Trace View** > **Data Math**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:MATH:FUNCtion {NORMal|SUBTract|DIVide|ADD|MULTiply|USER}

:CALCulate:PS[1-1]:TRACe[1-1]:MATH:FUNCtion?

Query Response

{NORMal|SUBTract|DIVide|ADD|MULTiply|USER} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:MATH:FUNCtion NORMal"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:MATH:FUNCtion?"
30 ENTER 717;A$
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).MATH.MEMorize

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).MATH.MEMorize

Description

This command copies data to memory, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PS1.TRACe1.MATH.MEMorize

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).MATH.FUNCtion

SCPI.CALCulate.PS(Ch).TRACe(Tr).MATH.OFFSet

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:MATH:MEMorize

Example of use

10 OUTPUT 717;":CALCulate:PS1:TRACe1:MATH:MEMorize"

SCPI.CALCulate.PS(Ch).TRACe(Tr).MATH.OFFSet

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).MATH.OFFSet = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).MATH.OFFSet

Description

This command sets/gets Trace Offset, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace Offset
Data Type	Double precision floating point type (Double)
Range	-500G ~ 500G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -500000000000
SCPI.CALCulate.PS1.TRACe1.MATH.OFFSet = Var
Var = SCPI.CALCulate.PS1.TRACe1.MATH.OFFSet
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).MATH.FUNCtion

SCPI.CALCulate.PS(Ch).TRACe(Tr).MATH.MEMorize

Equivalent Key

PS Menu: **Trace View** > **Offset**

PS Menu: **Trace View** > **Marker** > **-Offset**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:MATH:OFFSet {-500G ~ 500G}

:CALCulate:PS[1-1]:TRACe[1-1]:MATH:OFFSet?

Query Response

{-500G ~ 500G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:MATH:OFFSet -500000000000"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:MATH:OFFSet ?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).SMOothing.APERture

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).SMOothing.APERture = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).SMOothing.APERture

Description

This command sets/gets Smoothing aperture, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Smoothing aperture
Data Type	Double precision floating point type (Double)
Range	50m ~ 25
Preset Value	1.5
Unit	%
Resolution	10m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0.05
SCPI.CALCulate.PS1.TRACe1.SMOothing.APERture = Var
Var = SCPI.CALCulate.PS1.TRACe1.SMOothing.APERture
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).SMOothing.STATe

Equivalent Key

PS Menu: **Trace View** > **Aperture**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:SMOothing:APERture {50m ~ 25}

:CALCulate:PS[1-1]:TRACe[1-1]:SMOothing:APERture?

Query Response

{50m ~ 25} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:SMOothing:APERture 0.05"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:SMOothing:APERture ?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).SMOothing.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).SMOothing.STATe = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).SMOothing.STATe

Description

This command sets/gets Smoothing on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Smoothing (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.PS1.TRACe1.SMOothing.STATe = Var
Var = SCPI.CALCulate.PS1.TRACe1.SMOothing.STATe
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).SMOothing.APERture

Equivalent Key

PS Menu: **Trace View** > **Smoothing**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:SMOothing:STATe {ON|OFF|1|0}

:CALCulate:PS[1-1]:TRACe[1-1]:SMOothing:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:SMOothing:STATe 1"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:SMOothing:STATe?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OMISsion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OMISsion = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OMISsion

Description

This command sets/gets Spurious display omission ON/OFF, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious display omission
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.PS1.TRACe1.SPURious.OMISsion = Var
Var = SCPI.CALCulate.PS1.TRACe1.SPURious.OMISsion
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:SPURious:OMISsion {ON|OFF|1|0}

:CALCulate:PS[1-1]:TRACe[1-1]:SPURious:OMISsion?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:SPURious:OMISsion 1"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:SPURious:OMISsion?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

Description

This command clears threshold table, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PS1.TRACe1.SPURious.OSSPur.FLISt.CLEar

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:SPURious:OSSPur:FLISt:CLEar

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:SPURious:OSSPur:FLISt:CLEar"
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

Description

This command sets/gets Spurious Table for Omit Specified Spurious, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious table for omit specified spurious
Data Type	Long integer type (Long)
Range	1 ~ 100
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.PS1.TRACe1.SPURious.OSSPur.FLISt.COUNt = Var
Var = SCPI.CALCulate.PS1.TRACe1.SPURious.OSSPur.FLISt.COUNt
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:SPURious:OSSPur:FLISt:COUNt {1 ~ 100}

:CALCulate:PS[1-1]:TRACe[1-1]:SPURious:OSSPur:FLISt:COUNt?

Query Response

{1 ~ 100} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:SPURious:OSSPur:FLISt:COUNt 1 "
20 OUTPUT 717;":CALCulate:PS1:TRACe1:SPURious:OSSPur:FLISt:COUNt?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

Description

This command sets/gets threshold table, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Threshold table
Data Type	Variant type Array (Range)
Range	1...100
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 100) as Variant
SCPI.CALCulate.PS1.TRACe1.SPURious.OSSPur.FLISt.DATA = Var
Var = SCPI.CALCulate.PS1.TRACe1.SPURious.OSSPur.FLISt.DATA
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:SPURious:OSSPur:FLISt:DATA {1...100}

:CALCulate:PS[1-1]:TRACe[1-1]:SPURious:OSSPur:FLISt:DATA?

Query Response

{1...100} <newline><^END>

Example of use

```
10 Dim A(1:100)
20 OUTPUT 717;":CALCulate:PS1:TRACe1:SPURious:OSSPur:FLISt:DATA 1,100"
30 OUTPUT 717;":CALCulate:PS1:TRACe1:SPURious:OSSPur:FLISt:DATA ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

Description

This command sets/gets Specified Spurious display omission ON/OFF, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious display omission (ON/OFF)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.PS1.TRACe1.SPURious.OSSPur.OMISsion = Var
Var = SCPI.CALCulate.PS1.TRACe1.SPURious.OSSPur.OMISsion
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:SPURious:OSSPur:OMISsion {ON|OFF|1|0}

:CALCulate:PS[1-1]:TRACe[1-1]:SPURious:OSSPur:OMISsion?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:SPURious:OSSPur:OMISsion 1"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:SPURious:OSSPur:OMISsion?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.POWer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.POWer = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.POWer

Description

This command sets/gets Spurious display merging ON/OFF, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious display merging
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.PS1.TRACe1.SPURious.POWer = Var
Var = SCPI.CALCulate.PS1.TRACe1.SPURious.POWer
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:SPURious:POWer {ON|OFF|1|0}

:CALCulate:PS[1-1]:TRACe[1-1]:SPURious:POWer?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:SPURious:POWer 1"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:SPURious:POWer?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.SENSibility

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.SENSibility = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.SENSibility

Description

This command sets/gets spurious sensibility, for the selected trace *Tr* of the selected channel *Ch*.

The low peak has real sensibility when the value is small.

Variable

Parameter	*Value*
Description	Spurious sensibility
Data Type	Double precision floating point type (Double)
Range	10m ~ 10
Preset Value	3
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 5
SCPI.CALCulate.PS1.TRACe1.SPURious.SENSibility = Var
Var = SCPI.CALCulate.PS1.TRACe1.SPURious.SENSibility
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OMISsion

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.CLEar

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.COUNt

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.FLISt.DATA

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.OSSPur.OMISsion

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.POWer

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.SLISt

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

PS Menu: **Trace View** > **Spurious** > **Spur Sensibility**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:SPURious:SENSibility {10m ~ 10}

:CALCulate:PS[1-1]:TRACe[1-1]:SPURious:SENSibility?

Query Response

{10m ~ 10} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:SPURious:SENSibility 5"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:SPURious:SENSibility?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.SLISt

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.SLISt

Description

This command gets Spurious List, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious list
Data Type	Variant type Array (Range)
Range	1...0
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 0) as Variant
Var = SCPI.CALCulate.PS1.TRACe1.SPURious.SLISt
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:SPURious:SLISt?

Query Response

{1...0} <newline><^END>

Example of use

```
10 Dim A(1:0)
20 OUTPUT 717;":CALCulate:PS1:TRACe1:SPURious:SLISt?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

Description

This command sets/gets Spurious minimum level definition, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Spurious minimum level definition
Data Type	Double precision floating point type (Double)
Range	-500 ~ 500
Preset Value	-500
Unit	dBc
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.PS1.TRACe1.SPURious.THReshold.LEVel.MINimum = Var
Var = SCPI.CALCulate.PS1.TRACe1.SPURious.THReshold.LEVel.MINimum
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

PS Menu: **Trace View** > **Spurious** > **Minimum Spur Level**

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:SPURious:THReshold:LEVel:MINimum {-500 ~ 500}

:CALCulate:PS[1-1]:TRACe[1-1]:SPURious:THReshold:LEVel:MINimum?

Query Response

{-500 ~ 500} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:SPURious:THReshold:LEVel:MINimum 0"
20 OUTPUT 717;":CALCulate:PS1:TRACe1:SPURious:THReshold:LEVel:MINimum?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

Description

This command clears threshold table, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.PS1.TRACe1.SPURious.THReshold.TABLe.CLEar

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:CLEar

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:SPURious:THReshold:TABLe:CLEar"
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

Description

This command sets/gets number of threshold segments, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Number of threshold segments
Data Type	Long integer type (Long)
Range	1 ~ 20
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.PS1.TRACe1.SPURious.THReshold.TABLe.COUNt = Var
Var = SCPI.CALCulate.PS1.TRACe1.SPURious.THReshold.TABLe.COUNt
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:COUNt {1 ~ 20}

:CALCulate:PS[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:COUNt?

Query Response

{1 ~ 20} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:PS1:TRACe1:SPURious:THReshold:TABLe:COUNt 1 "
20 OUTPUT 717;":CALCulate:PS1:TRACe1:SPURious:THReshold:TABLe:COUNt?"
30 ENTER 717;A
```

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA = *Value*

Value = SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.TABLe.DATA

Description

This command sets/gets threshold table, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Threshold table
Data Type	Variant type Array (Range)
Range	1...60
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 60) as Variant
SCPI.CALCulate.PS1.TRACe1.SPURious.THReshold.TABLe.DATA = Var
Var = SCPI.CALCulate.PS1.TRACe1.SPURious.THReshold.TABLe.DATA
```

Related Objects

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.LEVel.MINimum

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.TABLe.CLEar

SCPI.CALCulate.PS(Ch).TRACe(Tr).SPURious.THReshold.TABLe.COUNt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:PS[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:DATA {1...60}

:CALCulate:PS[1-1]:TRACe[1-1]:SPURious:THReshold:TABLe:DATA?

Query Response

{1...60} <newline><^END>

Example of use

```
10 Dim A(1:60)
20 OUTPUT 717;":CALCulate:PS1:TRACe1:SPURious:THReshold:TABLe:DATA 1,60"
30 OUTPUT 717;":CALCulate:PS1:TRACe1:SPURious:THReshold:TABLe:DATA?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.SP(Ch).ALLTrace.LIMit.FAIL

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.SP(Ch).ALLTrace.LIMit.FAIL

Description

This command gets limit test pass/fail, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Boolean
Var= 1
Var = SCPI.CALCulate.SP1.ALLTrace.LIMit.FAIL
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:ALLTrace:LIMit:FAIL?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:ALLTrace:LIMit:FAIL ?"
20 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).ALLTrace.MARKer.COUPle.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).ALLTrace.MARKer.COUPle.STATe = *Value*

Value = SCPI.CALCulate.SP(Ch).ALLTrace.MARKer.COUPle.STATe

Description

This command sets/gets marker couple on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker Coupling Function state (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.SP1.ALLTrace.MARKer.COUPle.STATe = Var
Var = SCPI.CALCulate.SP1.ALLTrace.MARKer.COUPle.STATe
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:ALLTrace:MARKer:COUPle:STATe {ON|OFF|1|0}

:CALCulate:SP[1-1]:ALLTrace:MARKer:COUPle:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:ALLTrace:MARKer:COUPle:STATe 1"
20 OUTPUT 717;":CALCulate:SP1:ALLTrace:MARKer:COUPle:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).ALLTrace.MARKer.DISCrete.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).ALLTrace.MARKer.DISCrete.STATe = *Value*

Value = SCPI.CALCulate.SP(Ch).ALLTrace.MARKer.DISCrete.STATe

Description

This command sets/gets marker discrete on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Desc_Value
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.SP1.ALLTrace.MARKer.DISCrete.STATe = Var
Var = SCPI.CALCulate.SP1.ALLTrace.MARKer.DISCrete.STATe
```

Related Objects

Equivalent Key

SP Menu: **Marker** > **More Functions** > **Discrete**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:ALLTrace:MARKer:DISCrete:STATe {ON|OFF|1|0}

:CALCulate:SP[1-1]:ALLTrace:MARKer:DISCrete:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:ALLTrace:MARKer:DISCrete:STATe 1"
20 OUTPUT 717;":CALCulate:SP1:ALLTrace:MARKer:DISCrete:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).ALLTrace.MARKer.REFerence.NUMBer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).ALLTrace.MARKer.REFerence.NUMBer = *Value*

Value = SCPI.CALCulate.SP(Ch).ALLTrace.MARKer.REFerence.NUMBer

Description

This command sets/gets reference marker number, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Enables/Disables marker discrete
Data Type	Long integer type (Long)
Range	1 ~ 10
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.SP1.ALLTrace.MARKer.REFerence.NUMBer = Var
Var = SCPI.CALCulate.SP1.ALLTrace.MARKer.REFerence.NUMBer
```

Related Objects

SCPI.CALCulate.SP(Ch).ALLTrace.MARKer.REFerence.STATe

Equivalent Key

SP Menu: **Marker** > **More Functions** > **Ref Marker**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:ALLTrace:MARKer:REFerence:NUMBer {1 ~ 10}

:CALCulate:SP[1-1]:ALLTrace:MARKer:REFerence:NUMBer?

Query Response

{1 ~ 10} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:ALLTrace:MARKer:REFerence:NUMBer 1 "
20 OUTPUT 717;":CALCulate:SP1:ALLTrace:MARKer:REFerence:NUMBer ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).ALLTrace.MARKer.REFerence.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).ALLTrace.MARKer.REFerence.STATe = *Value*

Value = SCPI.CALCulate.SP(Ch).ALLTrace.MARKer.REFerence.STATe

Description

This command sets/gets delta marker mode on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker reference number
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.SP1.ALLTrace.MARKer.REFerence.STATe = Var
Var = SCPI.CALCulate.SP1.ALLTrace.MARKer.REFerence.STATe
```

Related Objects

SCPI.CALCulate.SP(Ch).ALLTrace.MARKer.REFerence.NUMBer

Equivalent Key

SP Menu: **Marker** > **More Functions** > **Ref Marker Mode**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:ALLTrace:MARKer:REFerence:STATe {ON|OFF|1|0}

:CALCulate:SP[1-1]:ALLTrace:MARKer:REFerence:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:ALLTrace:MARKer:REFerence:STATe 1"
20 OUTPUT 717;":CALCulate:SP1:ALLTrace:MARKer:REFerence:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).DATA.RDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).DATA.RDATa = *Value*

Value = SCPI.CALCulate.SP(Ch).DATA.RDATa

Description

This command sets/gets measurement raw data, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Measurement raw data
Data Type	Variant type Array (Range)
Range	1...1024
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1024) as Variant
SCPI.CALCulate.SP1.DATA.RDATa = Var
Var = SCPI.CALCulate.SP1.DATA.RDATa
```

Related Objects

SCPI.CALCulate.SP(Ch).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:DATA:RDATa {1...1024}

:CALCulate:SP[1-1]:DATA:RDATa?

Query Response

{1...1024} <newline><^END>

Example of use

```
10 Dim A(1:1024)
20 OUTPUT 717;":CALCulate:SP1:DATA:RDATa 1,1024"
30 OUTPUT 717;":CALCulate:SP1:DATA:RDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.SP(Ch).DATA.XDATa

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.SP(Ch).DATA.XDATa

Description

This command gets X axis data, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	X axis data
Data Type	Variant type Array (Range)
Range	1...1024
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1024) as Variant
Var = SCPI.CALCulate.SP1.DATA.XDATa
```

Related Objects

SCPI.CALCulate.SP(Ch).DATA.RDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:DATA:XDATa?

Query Response

{1...1024} <newline><^END>

Example of use

```
10 Dim A(1:1024)
20 OUTPUT 717;":CALCulate:SP1:DATA:XDATa ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).ALLMarker.ACTive

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).ALLMarker.ACTive = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).ALLMarker.ACTive

Description

This command sets/gets active marker, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Active marker
Data Type	Long integer type (Long)
Range	1 ~ 10
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.SP1.TRACe1.ALLMarker.ACTive = Var
Var = SCPI.CALCulate.SP1.TRACe1.ALLMarker.ACTive
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:ALLMarker:ACTive {1 ~ 10}

:CALCulate:SP[1-1]:TRACe[1-1]:ALLMarker:ACTive?

Query Response

{1 ~ 10} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:ALLMarker:ACTive 1 "
20 OUTPUT 717;":CALCulate:SP1:TRACe1:ALLMarker:ACTive ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

Description

This command sets/gets marker search X range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search X range source
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	FRANge
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.SP1.TRACe1.ALLMarker.SEARch.DOMain.X = Var
Var = SCPI.CALCulate.SP1.TRACe1.ALLMarker.SEARch.DOMain.X
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

SCPI.CALCulate.SP(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Equivalent Key

SP Menu: **Marker Search** > **Search Range (X)**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:ALLMarker:SEARch:DOMain:X {FRANge|BDMarker}

:CALCulate:SP[1-1]:TRACe[1-1]:ALLMarker:SEARch:DOMain:X?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:ALLMarker:SEARch:DOMain:X FRANge"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:ALLMarker:SEARch:DOMain:X ?"
30 ENTER 717;A$
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Description

This command sets/gets marker search Y range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search Y range
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	FRANge
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.SP1.TRACe1.ALLMarker.SEARch.DOMain.Y = Var
Var = SCPI.CALCulate.SP1.TRACe1.ALLMarker.SEARch.DOMain.Y
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

SCPI.CALCulate.SP(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Equivalent Key

SP Menu: **Marker Search** > **Search Range (Y)**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:ALLMarker:SEARch:DOMain:Y {FRANge|BDMarker}

:CALCulate:SP[1-1]:TRACe[1-1]:ALLMarker:SEARch:DOMain:Y?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:ALLMarker:SEARch:DOMain:Y FRANge"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:ALLMarker:SEARch:DOMain:Y ?"
30 ENTER 717;A$
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Description

This command sets search peak all, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.SP1.TRACe1.ALLMarker.SEARch.PEAK

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

SCPI.CALCulate.SP(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Equivalent Key

SP Menu: **Marker Search** > **Peak** > **Search Peak All**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:ALLMarker:SEARch:PEAK

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:ALLMarker:SEARch:PEAK"
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.CENTer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.CENTer = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.CENTer

Description

This command sets/gets band marker X center, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X center
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.SP1.TRACe1.BDMarker.X.CENTer = Var
Var = SCPI.CALCulate.SP1.TRACe1.BDMarker.X.CENTer
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

SP Menu: **Marker Search** > **Band Marker X** > **Center**

SP Menu: **Marker Function** > **Band Marker X** > **Center**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:BDMarker:X:CENTer {-1T ~ 1T}

:CALCulate:SP[1-1]:TRACe[1-1]:BDMarker:X:CENTer?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:BDMarker:X:CENTer -10000000000"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:BDMarker:X:CENTer ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.SPAN

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.SPAN = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.SPAN

Description

This command sets/gets band marker X span, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X span
Data Type	Double precision floating point type (Double)
Range	0 ~ 2T
Preset Value	2T
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.SP1.TRACe1.BDMarker.X.SPAN = Var
Var = SCPI.CALCulate.SP1.TRACe1.BDMarker.X.SPAN
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

SP Menu: **Marker Search** > **Band Marker X** > **Span**

SP Menu: **Marker Function** > **Band Marker X** > **Span**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:BDMarker:X:SPAN {0 ~ 2T}

:CALCulate:SP[1-1]:TRACe[1-1]:BDMarker:X:SPAN?

Query Response

{0 ~ 2T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:BDMarker:X:SPAN 0"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:BDMarker:X:SPAN ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.STARt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.STARt = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.STARt

Description

This command sets/gets band marker X start, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X start
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	-1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.SP1.TRACe1.BDMarker.X.STARt = Var
Var = SCPI.CALCulate.SP1.TRACe1.BDMarker.X.STARt
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

SP Menu: **Marker Search** > **Band Marker X** > **Start**

SP Menu: **Marker Function** > **Band Marker X** > **Start**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:BDMarker:X:STARt {-1T ~ 1T}

:CALCulate:SP[1-1]:TRACe[1-1]:BDMarker:X:STARt?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:BDMarker:X:STARt -10000000000"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:BDMarker:X:STARt ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.STATe = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.STATe

Description

This command sets/gets band marker visible on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker visibility (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.SP1.TRACe1.BDMarker.X.STATe = Var
Var = SCPI.CALCulate.SP1.TRACe1.BDMarker.X.STATe
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

SP Menu: **Marker Search** > **Band Marker X** > **Band Marker X**

SP Menu: **Marker Function** > **Band Marker X** > **Band Marker X**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:BDMarker:X:STATe {ON|OFF|1|0}

:CALCulate:SP[1-1]:TRACe[1-1]:BDMarker:X:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:BDMarker:X:STATe 1"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:BDMarker:X:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.STOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.STOP = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.STOP

Description

This command sets/gets band marker X stop, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X stop
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.SP1.TRACe1.BDMarker.X.STOP = Var
Var = SCPI.CALCulate.SP1.TRACe1.BDMarker.X.STOP
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.X.STATe

Equivalent Key

SP Menu: **Marker Search** > **Band Marker X** > **Stop**

SP Menu: **Marker Function** > **Band Marker X** > **Stop**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:BDMarker:X:STOP {-1T ~ 1T}

:CALCulate:SP[1-1]:TRACe[1-1]:BDMarker:X:STOP?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:BDMarker:X:STOP -10000000000"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:BDMarker:X:STOP ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.CENTer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.CENTer = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.CENTer

Description

This command sets/gets band marker Y center, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y center
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.SP1.TRACe1.BDMarker.Y.CENTer = Var
Var = SCPI.CALCulate.SP1.TRACe1.BDMarker.Y.CENTer
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

SP Menu: **Marker Search** > **Band Marker Y** > **Center**

SP Menu: **Marker Function** > **Band Marker Y** > **Center**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:BDMarker:Y:CENTer {-1T ~ 1T}

:CALCulate:SP[1-1]:TRACe[1-1]:BDMarker:Y:CENTer?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:BDMarker:Y:CENTer -10000000000"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:BDMarker:Y:CENTer ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.SPAN

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.SPAN = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.SPAN

Description

This command sets/gets band marker Y span, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y span
Data Type	Double precision floating point type (Double)
Range	0 ~ 2T
Preset Value	2T
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.SP1.TRACe1.BDMarker.Y.SPAN = Var
Var = SCPI.CALCulate.SP1.TRACe1.BDMarker.Y.SPAN
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

SP Menu: **Marker Search** > **Band Marker Y** > **Span**

SP Menu: **Marker Function** > **Band Marker Y** > **Span**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:BDMarker:Y:SPAN {0 ~ 2T}

:CALCulate:SP[1-1]:TRACe[1-1]:BDMarker:Y:SPAN?

Query Response

{0 ~ 2T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:BDMarker:Y:SPAN 0"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:BDMarker:Y:SPAN ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.STARt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.STARt = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.STARt

Description

This command sets/gets band marker Y start, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y start
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	-1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.SP1.TRACe1.BDMarker.Y.STARt = Var
Var = SCPI.CALCulate.SP1.TRACe1.BDMarker.Y.STARt
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

SP Menu: **Marker Search** > **Band Marker Y** > **Start**

SP Menu: **Marker Function** > **Band Marker Y** > **Start**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:BDMarker:Y:STARt {-1T ~ 1T}

:CALCulate:SP[1-1]:TRACe[1-1]:BDMarker:Y:STARt?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:BDMarker:Y:STARt -10000000000"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:BDMarker:Y:STARt ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.STATe = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.STATe

Description

This command sets/gets band marker visible on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y state
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.SP1.TRACe1.BDMarker.Y.STATe = Var
Var = SCPI.CALCulate.SP1.TRACe1.BDMarker.Y.STATe
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

SP Menu: **Marker Search** > **Band Marker Y** > **Band Marker Y**

SP Menu: **Marker Function** > **Band Marker Y** > **Band Marker Y**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:BDMarker:Y:STATe {ON|OFF|1|0}

:CALCulate:SP[1-1]:TRACe[1-1]:BDMarker:Y:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:BDMarker:Y:STATe 1"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:BDMarker:Y:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.STOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.STOP = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.STOP

Description

This command sets/gets band marker Y stop, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y stop
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.SP1.TRACe1.BDMarker.Y.STOP = Var
Var = SCPI.CALCulate.SP1.TRACe1.BDMarker.Y.STOP
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.SP(Ch).TRACe(Tr).BDMarker.Y.STATe

Equivalent Key

SP Menu: **Marker Search** > **Band Marker Y** > **Stop**

SP Menu: **Marker Function** > **Band Marker Y** > **Stop**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:BDMarker:Y:STOP {-1T ~ 1T}

:CALCulate:SP[1-1]:TRACe[1-1]:BDMarker:Y:STOP?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:BDMarker:Y:STOP -10000000000"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:BDMarker:Y:STOP ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.COPY

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.COPY = *Value*

Description

This command copies data to user trace, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Copies date to user trace
Data Type	Long integer type (Long)
Range	1 ~ 8
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.SP1.TRACe1.DATA.COPY = Var
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:DATA:COPY {1 ~ 8}

Query Response

{1 ~ 8} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:DATA:COPY 1 "
20 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.FDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.FDATa = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.FDATa

Description

This command sets/gets formatted trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Formatted trace data
Data Type	Variant type Array (Range)
Range	1...1024
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1024) as Variant
SCPI.CALCulate.SP1.TRACe1.DATA.FDATa = Var
Var = SCPI.CALCulate.SP1.TRACe1.DATA.FDATa
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:DATA:FDATa {1...1024}

:CALCulate:SP[1-1]:TRACe[1-1]:DATA:FDATa?

Query Response

{1...1024} <newline><^END>

Example of use

```
10 Dim A(1:1024)
20 OUTPUT 717;":CALCulate:SP1:TRACe1:DATA:FDATa 1,1024"
30 OUTPUT 717;":CALCulate:SP1:TRACe1:DATA:FDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.FMEMory

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.FMEMory = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.FMEMory

Description

This command sets/gets formatted memory data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Formatted memory data
Data Type	Variant type Array (Range)
Range	1...1024
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1024) as Variant
SCPI.CALCulate.SP1.TRACe1.DATA.FMEMory = Var
Var = SCPI.CALCulate.SP1.TRACe1.DATA.FMEMory
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:DATA:FMEMory {1...1024}

:CALCulate:SP[1-1]:TRACe[1-1]:DATA:FMEMory?

Query Response

{1...1024} <newline><^END>

Example of use

```
10 Dim A(1:1024)
20 OUTPUT 717;":CALCulate:SP1:TRACe1:DATA:FMEMory 1,1024"
30 OUTPUT 717;":CALCulate:SP1:TRACe1:DATA:FMEMory ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.UDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.UDATa = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.UDATa

Description

This command sets/gets unformatted trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Unformatted trace data
Data Type	Variant type Array (Range)
Range	1...1024
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1024) as Variant
SCPI.CALCulate.SP1.TRACe1.DATA.UDATa = Var
Var = SCPI.CALCulate.SP1.TRACe1.DATA.UDATa
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:DATA:UDATa {1...1024}

:CALCulate:SP[1-1]:TRACe[1-1]:DATA:UDATa?

Query Response

{1...1024} <newline><^END>

Example of use

```
10 Dim A(1:1024)
20 OUTPUT 717;":CALCulate:SP1:TRACe1:DATA:UDATa 1,1024"
30 OUTPUT 717;":CALCulate:SP1:TRACe1:DATA:UDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.UMEMory

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.UMEMory = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.UMEMory

Description

This command sets/gets unformatted memory data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Unformatted memory data
Data Type	Variant type Array (Range)
Range	1...1024
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1024) as Variant
SCPI.CALCulate.SP1.TRACe1.DATA.UMEMory = Var
Var = SCPI.CALCulate.SP1.TRACe1.DATA.UMEMory
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.SP(Ch).TRACe(Tr).DATA.UDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:DATA:UMEMory {1...1024}

:CALCulate:SP[1-1]:TRACe[1-1]:DATA:UMEMory?

Query Response

{1...1024} <newline><^END>

Example of use

```
10 Dim A(1:1024)
20 OUTPUT 717;":CALCulate:SP1:TRACe1:DATA:UMEMory 1,1024"
30 OUTPUT 717;":CALCulate:SP1:TRACe1:DATA:UMEMory ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).EQUation.TEXT

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).EQUation.TEXT = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).EQUation.TEXT

Description

This command sets/gets equation string, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Equation string
Data Type	Character string type (String)
Range	254 chars
Preset Value	""
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	170, "Invalid equation label" 171, "Invalid equation

	expression"

Examples

```
Dim Var as String
SCPI.CALCulate.SP1.TRACe1.EQUation.TEXT = Var
Var = SCPI.CALCulate.SP1.TRACe1.EQUation.TEXT
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:EQUation:TEXT

:CALCulate:SP[1-1]:TRACe[1-1]:EQUation:TEXT?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:EQUation:TEXT Sample_Text"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:EQUation:TEXT ?"
30 ENTER 717;A$
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).FORMat

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).FORMat = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).FORMat

Description

This command sets/gets SP format, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	SP format
Data Type	Character string type (String)
Range	DBM|DBV|WATT|VOLT|DBMHz|DBVHz|WHZ|VHZ
Preset Value	DBM
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "DBM"
SCPI.CALCulate.SP1.TRACe1.FORMat = Var
Var = SCPI.CALCulate.SP1.TRACe1.FORMat
```

Equivalent Key

SP Menu: **Format** > **Format**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:FORMat
{DBM|DBV|WATT|VOLT|DBMHz|DBVHz|WHZ|VHZ}

:CALCulate:SP[1-1]:TRACe[1-1]:FORMat?

Query Response

{DBM|DBV|WATT|VOLT|DBMHz|DBVHz|WHZ|VHZ} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:FORMat DBM"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:FORMat ?"
30 ENTER 717;A$
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).FUNCtion.DOMain.X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).FUNCtion.DOMain.X = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).FUNCtion.DOMain.X

Description

This command sets/gets X analysis range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	X analysis range source
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	BDMarker
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.SP1.TRACe1.FUNCtion.DOMain.X = Var
Var = SCPI.CALCulate.SP1.TRACe1.FUNCtion.DOMain.X
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).FUNCtion.DOMain.Y

Equivalent Key

SP Menu: **Marker Function** > **Analysis Range (X)**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:FUNCtion:DOMain:X {FRANge|BDMarker}

:CALCulate:SP[1-1]:TRACe[1-1]:FUNCtion:DOMain:X?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:FUNCtion:DOMain:X FRANge"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:FUNCtion:DOMain:X ?"
30 ENTER 717;A$
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).FUNCtion.DOMain.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).FUNCtion.DOMain.Y = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).FUNCtion.DOMain.Y

Description

This command sets/gets Y analysis range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Y analysis range source
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	BDMarker
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.SP1.TRACe1.FUNCtion.DOMain.Y = Var
Var = SCPI.CALCulate.SP1.TRACe1.FUNCtion.DOMain.Y
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).FUNCtion.DOMain.X

Equivalent Key

SP Menu: **Marker Function** > **Analysis Range (Y)**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:FUNCtion:DOMain:Y {FRANge|BDMarker}

:CALCulate:SP[1-1]:TRACe[1-1]:FUNCtion:DOMain:Y?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:FUNCtion:DOMain:Y FRANge"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:FUNCtion:DOMain:Y ?"
30 ENTER 717;A$
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak

Description

This command gets trace data statistics, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*mean*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*std_dev*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*peak_to_peak*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-

Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
Var = SCPI.CALCulate.SP1.TRACe1.FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:FUNCtion:STATistics:DATA?

Query Response

{ - - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:FUNCtion:STATistics:DATA ?"
30 ENTER 717;A$
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak

Description

This command gets memory data statistics, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*mean*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*std_dev*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*peak_to_peak*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-

Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
Var = SCPI.CALCulate.SP1.TRACe1.FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).FUNCtion.STATistics.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:FUNCtion:STATistics:MEMory?

Query Response

{ - - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:FUNCtion:STATistics:MEMory ?"
30 ENTER 717;A$
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).FUNCtion.TYPE

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).FUNCtion.TYPE = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).FUNCtion.TYPE

Description

This command sets/gets analysis type, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Analysis type
Data Type	Character string type (String)
Range	OFF|STATistics
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.SP1.TRACe1.FUNCtion.TYPE = Var
Var = SCPI.CALCulate.SP1.TRACe1.FUNCtion.TYPE
```

Equivalent Key

SP Menu: **Marker Function** > **Analysis Type**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:FUNCtion:TYPE {OFF|STATistics}

:CALCulate:SP[1-1]:TRACe[1-1]:FUNCtion:TYPE?

Query Response

{OFF|STATistics} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:FUNCtion:TYPE OFF"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:FUNCtion:TYPE ?"
30 ENTER 717;A$
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).HOLD

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).HOLD = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).HOLD

Description

This command sets/gets Data hold, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Data hold
Data Type	Character string type (String)
Range	OFF|MAXimum|MINimum
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.SP1.TRACe1.HOLD = Var
Var = SCPI.CALCulate.SP1.TRACe1.HOLD
```

Equivalent Key

SP Menu: **Trace View** > **Data Hold**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:HOLD {OFF|MAXimum|MINimum}

:CALCulate:SP[1-1]:TRACe[1-1]:HOLD?

Query Response

{OFF|MAXimum|MINimum} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:HOLD OFF"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:HOLD ?"
30 ENTER 717;A$
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.FAIL

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.FAIL

Description

This command gets limit test pass/fail, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test pass/fail
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Boolean
Var= 1
Var = SCPI.CALCulate.SP1.TRACe1.LIMit.FAIL
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:LIMit:FAIL?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:LIMit:FAIL ?"
20 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.LOWer.LDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.LOWer.LDATa = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.LOWer.LDATa

Description

This command sets/gets expanded lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Expanded lower limit line
Data Type	Variant type Array (Range)
Range	1...1024
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1024) as Variant
SCPI.CALCulate.SP1.TRACe1.LIMit.LOWer.LDATa = Var
Var = SCPI.CALCulate.SP1.TRACe1.LIMit.LOWer.LDATa
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:LIMit:LOWer:LDATa {1...1024}

:CALCulate:SP[1-1]:TRACe[1-1]:LIMit:LOWer:LDATa?

Query Response

{1...1024} <newline><^END>

Example of use

```
10 Dim A(1:1024)
20 OUTPUT 717;":CALCulate:SP1:TRACe1:LIMit:LOWer:LDATa 1,1024"
30 OUTPUT 717;":CALCulate:SP1:TRACe1:LIMit:LOWer:LDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

Description

This command clears lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.SP1.TRACe1.LIMit.LOWer.SEGMent.CLEar

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:CLEar

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:LIMit:LOWer:SEGMent:CLEar"
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

Description

This command sets/gets number of lower limit segments, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Number of lower limit segments
Data Type	Long integer type (Long)
Range	1 ~ 100
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.SP1.TRACe1.LIMit.LOWer.SEGMent.COUNt = Var
Var = SCPI.CALCulate.SP1.TRACe1.LIMit.LOWer.SEGMent.COUNt
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:COUNt {1 ~ 100}

:CALCulate:SP[1-1]:TRACe[1-1]:LIMit:LOWer:SEGMent:COUNt?

Query Response

{1 ~ 100} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:LIMit:LOWer:SEGMent:COUNt 1 "
20 OUTPUT 717;":CALCulate:SP1:TRACe1:LIMit:LOWer:SEGMent:COUNt ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Description

This command sets/gets lower limit line data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Lower limit line data
Data Type	Variant type Array (Range)
Range	1...400
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 400) as Variant
SCPI.CALCulate.SP1.TRACe1.LIMit.LOWer.SEGment.DATA = Var
Var = SCPI.CALCulate.SP1.TRACe1.LIMit.LOWer.SEGment.DATA
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.LOWer.SEGment.CLEar

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.LOWer.SEGment.COUNt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:LIMit:LOWer:SEGment:DATA {1...400}

:CALCulate:SP[1-1]:TRACe[1-1]:LIMit:LOWer:SEGment:DATA?

Query Response

{1...400} <newline><^END>

Example of use

```
10 Dim A(1:400)
20 OUTPUT 717;":CALCulate:SP1:TRACe1:LIMit:LOWer:SEGment:DATA 1,400"
30 OUTPUT 717;":CALCulate:SP1:TRACe1:LIMit:LOWer:SEGment:DATA ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.REPort.DATA

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.REPort.DATA

Description

This command gets pass/fail list, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Pass/fail list
Data Type	Variant type Array (Range)
Range	1...1024
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1024) as Variant
Var = SCPI.CALCulate.SP1.TRACe1.LIMit.REPort.DATA
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:LIMit:REPort[:DATA]?

Query Response

{1...1024} <newline><^END>

Example of use

```
10 Dim A(1:1024)
20 OUTPUT 717;":CALCulate:SP1:TRACe1:LIMit:REPort[:DATA] ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.STATe = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.STATe

Description

This command sets/gets limit test on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.SP1.TRACe1.LIMit.STATe = Var
Var = SCPI.CALCulate.SP1.TRACe1.LIMit.STATe
```

Equivalent Key

SP Menu: **Display** > **Limit Test** > **Limit Test**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:LIMit[:STATe] {ON|OFF|1|0}

:CALCulate:SP[1-1]:TRACe[1-1]:LIMit[:STATe]?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:LIMit[:STATe] 1"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:LIMit[:STATe] ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.UPPer.LDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.UPPer.LDATa = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.UPPer.LDATa

Description

This command sets/gets expanded upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Expanded lower limit line
Data Type	Variant type Array (Range)
Range	1...1024
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1024) as Variant
SCPI.CALCulate.SP1.TRACe1.LIMit.UPPer.LDATa = Var
Var = SCPI.CALCulate.SP1.TRACe1.LIMit.UPPer.LDATa
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:LIMit:UPPer:LDATa {1...1024}

:CALCulate:SP[1-1]:TRACe[1-1]:LIMit:UPPer:LDATa?

Query Response

{1...1024} <newline><^END>

Example of use

```
10 Dim A(1:1024)
20 OUTPUT 717;":CALCulate:SP1:TRACe1:LIMit:UPPer:LDATa 1,1024"
30 OUTPUT 717;":CALCulate:SP1:TRACe1:LIMit:UPPer:LDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

Description

This command clears upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.SP1.TRACe1.LIMit.UPPer.SEGMent.CLEar

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:CLEar

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:LIMit:UPPer:SEGMent:CLEar"
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Description

This command sets/gets number of upper limit segments, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Number of upper limit segments
Data Type	Long integer type (Long)
Range	1 ~ 100
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.SP1.TRACe1.LIMit.UPPer.SEGMent.COUNt = Var
Var = SCPI.CALCulate.SP1.TRACe1.LIMit.UPPer.SEGMent.COUNt
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:COUNt {1 ~ 100}

:CALCulate:SP[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:COUNt?

Query Response

{1 ~ 100} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:LIMit:UPPer:SEGMent:COUNt 1 "
20 OUTPUT 717;":CALCulate:SP1:TRACe1:LIMit:UPPer:SEGMent:COUNt ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Description

This command sets/gets upper limit line data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Upper limit line data
Data Type	Variant type Array (Range)
Range	1...400
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 400) as Variant
SCPI.CALCulate.SP1.TRACe1.LIMit.UPPer.SEGMent.DATA = Var
Var = SCPI.CALCulate.SP1.TRACe1.LIMit.UPPer.SEGMent.DATA
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.SP(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:DATA {1...400}

:CALCulate:SP[1-1]:TRACe[1-1]:LIMit:UPPer:SEGMent:DATA?

Query Response

{1...400} <newline><^END>

Example of use

```
10 Dim A(1:400)
20 OUTPUT 717;":CALCulate:SP1:TRACe1:LIMit:UPPer:SEGMent:DATA 1,400"
30 OUTPUT 717;":CALCulate:SP1:TRACe1:LIMit:UPPer:SEGMent:DATA ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

Description

This command sets marker search LPEak, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.SP1.TRACe1.MARKer10.SEARch.EXECute.LPEak

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

Equivalent Key

SP Menu: **Marker Search** > **Peak** > **Search Left**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:LPEak

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:MARKer10:SEARch:EXECute:LPEak"
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

Description

This command sets marker search LTARget, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.SP1.TRACe1.MARKer10.SEARch.EXECute.LTARget

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

Equivalent Key

SP Menu: **Marker Search** > **Target** > **Search Left**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:LTARget

Example of use

10 OUTPUT 717;":CALCulate:SP1:TRACe1:MARKer10:SEARch:EXECute:LTARget"

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

Description

This command sets marker search MAXimum, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.SP1.TRACe1.MARKer10.SEARch.EXECute.MAXimum

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

Equivalent Key

SP Menu: **Marker Search** > **Search Max**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:MAXimum

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:MARKer10:SEARch:EXECute:MAXimum"
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

Description

This command sets marker search MINimum, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.SP1.TRACe1.MARKer10.SEARch.EXECute.MINimum

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

Equivalent Key

SP Menu: **Marker Search** > **Search Min**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:MINimum

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:MARKer10:SEARch:EXECute:MINimum"
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

Description

This command sets marker search PEAK, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.SP1.TRACe1.MARKer10.SEARch.EXECute.PEAK

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

Equivalent Key

SP Menu: **Marker Search** > **Peak** > **Search Peak**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:PEAK

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:MARKer10:SEARch:EXECute:PEAK"
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

Description

This command sets marker search RPEak, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.SP1.TRACe1.MARKer10.SEARch.EXECute.RPEak

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

Equivalent Key

SP Menu: **Marker Search** > **Peak** > **Search Right**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:RPEak

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:MARKer10:SEARch:EXECute:RPEak"
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

Description

This command sets marker search RTARget, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.SP1.TRACe1.MARKer10.SEARch.EXECute.RTARget

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

SP Menu: **Marker Search** > **Target** > **Search Right**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:RTARget

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:MARKer10:SEARch:EXECute:RTARget"
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Description

This command sets marker search TARGet, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.SP1.TRACe1.MARKer10.SEARch.EXECute.TARGet

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

Equivalent Key

SP Menu: **Marker Search** > **Target** > **Search Target**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:EXECute:TARGet

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:MARKer10:SEARch:EXECute:TARGet"
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion

Description

This command sets/gets marker-search-peak excursion value, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker-search-peak excursion value
Data Type	Double precision floating point type (Double)
Range	0 ~ 10G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.SP1.TRACe1.MARKer10.SEARch.PEAK.EXCursion = Var
Var = SCPI.CALCulate.SP1.TRACe1.MARKer10.SEARch.PEAK.EXCursion
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity

Equivalent Key

SP Menu: **Marker Search** > **Peak** > **Peak Excursion**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:PEAK:EXCursion {0 ~ 10G}

:CALCulate:SP[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:PEAK:EXCursion?

Query Response

{0 ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:MARKer10:SEARch:PEAK:EXCursion 0"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:MARKer10:SEARch:PEAK:EXCursion ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity

Description

This command sets/gets marker-search-peak polarity type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker-search-peak polarity type
Data Type	Character string type (String)
Range	POSitive|NEGative|BOTH
Preset Value	POSitive
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "POSitive"
SCPI.CALCulate.SP1.TRACe1.MARKer10.SEARch.PEAK.POLarity = Var
Var = SCPI.CALCulate.SP1.TRACe1.MARKer10.SEARch.PEAK.POLarity
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion

Equivalent Key

SP Menu: **Marker Search** > **Peak** > **Peak Polarity**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:PEAK:POLarity {POSitive|NEGative|BOTH}

:CALCulate:SP[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:PEAK:POLarity?

Query Response

{POSitive|NEGative|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:MARKer10:SEARch:PEAK:POLarity POSitive"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:MARKer10:SEARch:PEAK:POLarity ?"
30 ENTER 717;A$
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition

Description

This command sets/gets marker-target transition type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker-target transition type
Data Type	Character string type (String)
Range	POSitive|NEGative|BOTH
Preset Value	BOTH
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as String
Var= "POSitive"
SCPI.CALCulate.SP1.TRACe1.MARKer10.SEARch.TARGet.TRANsition = Var
Var = SCPI.CALCulate.SP1.TRACe1.MARKer10.SEARch.TARGet.TRANsition
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y

Equivalent Key

SP Menu: **Marker Search** > **Target** > **Target Transition**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TARGet:TRANsition {POSitive|NEGative|BOTH}

:CALCulate:SP[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TARGet:TRANsition?

Query Response

{POSitive|NEGative|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:MARKer10:SEARch:TARGet:TRANsition POSitive"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:MARKer10:SEARch:TARGet:TRANsition ?"
30 ENTER 717;A$
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y

Description

This command sets/gets marker target value, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker target value
Data Type	Double precision floating point type (Double)
Range	-10G ~ 10G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.SP1.TRACe1.MARKer10.SEARch.TARGet.Y = Var
Var = SCPI.CALCulate.SP1.TRACe1.MARKer10.SEARch.TARGet.Y
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition

Equivalent Key

SP Menu: **Marker Search** > **Target** > **Target Value**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TARGet:Y {-10G ~ 10G}

:CALCulate:SP[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TARGet:Y?

Query Response

{-10G ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:MARKer10:SEARch:TARGet:Y -10000000000"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:MARKer10:SEARch:TARGet:Y ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE

Description

This command sets/gets search tracking type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Search tracking type
Data Type	Character string type (String)
Range	OFF|MAXimum|MINimum|PEAK|TARGet
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.SP1.TRACe1.MARKer10.SEARch.TRACking.TYPE = Var
Var = SCPI.CALCulate.SP1.TRACe1.MARKer10.SEARch.TRACking.TYPE
```

Equivalent Key

SP Menu: **Marker Search** > **Tracking**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TRACking:TYPE {OFF|MAXimum|MINimum|PEAK|TARGet}

:CALCulate:SP[1-1]:TRACe[1-1]:MARKer[1-10]:SEARch:TRACking:TYPE?

Query Response

{OFF|MAXimum|MINimum|PEAK|TARGet} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:MARKer10:SEARch:TRACking:TYPE OFF"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:MARKer10:SEARch:TRACking:TYPE ?"
30 ENTER 717;A$
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).STATe = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).STATe

Description

This command sets/gets marker visible on/off, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker visible (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.SP1.TRACe1.MARKer10.STATe = Var
Var = SCPI.CALCulate.SP1.TRACe1.MARKer10.STATe
```

Equivalent Key

SP Menu: **Marker** > **Clear Marker Menu** > **Marker 1**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:MARKer[1-10]:STATe {ON|OFF|1|0}

:CALCulate:SP[1-1]:TRACe[1-1]:MARKer[1-10]:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:MARKer10:STATe 1"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:MARKer10:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).X = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).X

Description

This command sets/gets marker x position, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker x position
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	992.5M
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
SCPI.CALCulate.SP1.TRACe1.MARKer10.X = Var
Var = SCPI.CALCulate.SP1.TRACe1.MARKer10.X
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).Y

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:MARKer[1-10]:X

:CALCulate:SP[1-1]:TRACe[1-1]:MARKer[1-10]:X?

Query Response

{Double} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:MARKer10:X 0"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:MARKer10:X ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).Y

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).Y

Description

This command gets marker y position, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker y position
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var = SCPI.CALCulate.SP1.TRACe1.MARKer10.Y
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).MARKer(Mk).X

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:MARKer[1-10]:Y?

Query Response

{Double} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:MARKer10:Y ?"
20 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).MATH.FUNCtion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).MATH.FUNCtion = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).MATH.FUNCtion

Description

This command sets/gets math operation type, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Math operation type
Data Type	Character string type (String)
Range	NORMal|SUBTract|DIVide|ADD|MULTiply|USER
Preset Value	NORMal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NORMal"
SCPI.CALCulate.SP1.TRACe1.MATH.FUNCtion = Var
Var = SCPI.CALCulate.SP1.TRACe1.MATH.FUNCtion
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).MATH.MEMorize

SCPI.CALCulate.SP(Ch).TRACe(Tr).MATH.OFFSet

Equivalent Key

SP Menu: **Trace View** > **Data Math**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:MATH:FUNCtion
{NORMal|SUBTract|DIVide|ADD|MULTiply|USER}

:CALCulate:SP[1-1]:TRACe[1-1]:MATH:FUNCtion?

Query Response

{NORMal|SUBTract|DIVide|ADD|MULTiply|USER} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:MATH:FUNCtion NORMal"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:MATH:FUNCtion ?"
30 ENTER 717;A$
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).MATH.MEMorize

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).MATH.MEMorize

Description

This command copies data to memory, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.SP1.TRACe1.MATH.MEMorize

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).MATH.FUNCtion

SCPI.CALCulate.SP(Ch).TRACe(Tr).MATH.OFFSet

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:MATH:MEMorize

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:MATH:MEMorize"
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).MATH.OFFSet

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).MATH.OFFSet = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).MATH.OFFSet

Description

This command sets/gets Trace Offset, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace Offset
Data Type	Double precision floating point type (Double)
Range	-500G ~ 500G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -500000000000
SCPI.CALCulate.SP1.TRACe1.MATH.OFFSet = Var
Var = SCPI.CALCulate.SP1.TRACe1.MATH.OFFSet
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).MATH.FUNCtion

SCPI.CALCulate.SP(Ch).TRACe(Tr).MATH.MEMorize

Equivalent Key

SP Menu: **Trace View** > **Offset**

SP Menu: **Trace View** > **Marker** > **-Offset**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:MATH:OFFSet {-500G ~ 500G}

:CALCulate:SP[1-1]:TRACe[1-1]:MATH:OFFSet?

Query Response

{-500G ~ 500G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:MATH:OFFSet -500000000000"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:MATH:OFFSet ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).SMOothing.APERture

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).SMOothing.APERture = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).SMOothing.APERture

Description

This command sets/gets Smoothing aperture, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Smoothing aperture
Data Type	Double precision floating point type (Double)
Range	50m ~ 25
Preset Value	1.5
Unit	%
Resolution	10m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0.05
SCPI.CALCulate.SP1.TRACe1.SMOothing.APERture = Var
Var = SCPI.CALCulate.SP1.TRACe1.SMOothing.APERture
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).SMOothing.STATe

Equivalent Key

SP Menu: **Trace View** > **Aperture**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:SMOothing:APERture {50m ~ 25}

:CALCulate:SP[1-1]:TRACe[1-1]:SMOothing:APERture?

Query Response

{50m ~ 25} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:SMOothing:APERture 0.05"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:SMOothing:APERture ?"
30 ENTER 717;A
```

SCPI.CALCulate.SP(Ch).TRACe(Tr).SMOothing.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.SP(Ch).TRACe(Tr).SMOothing.STATe = *Value*

Value = SCPI.CALCulate.SP(Ch).TRACe(Tr).SMOothing.STATe

Description

This command sets/gets Smoothing on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Desc_Value
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.SP1.TRACe1.SMOothing.STATe = Var
Var = SCPI.CALCulate.SP1.TRACe1.SMOothing.STATe
```

Related Objects

SCPI.CALCulate.SP(Ch).TRACe(Tr).SMOothing.APERture

Equivalent Key

SP Menu: **Trace View** > **Smoothing**

Equivalent SCPI Command

Syntax

:CALCulate:SP[1-1]:TRACe[1-1]:SMOothing:STATe {ON|OFF|1|0}

:CALCulate:SP[1-1]:TRACe[1-1]:SMOothing:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:SP1:TRACe1:SMOothing:STATe 1"
20 OUTPUT 717;":CALCulate:SP1:TRACe1:SMOothing:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).ALLTrace.ACTive

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).ALLTrace.ACTive = *Value*

Value = SCPI.CALCulate.TR(Ch).ALLTrace.ACTive

Description

This command sets/gets active trace, for the selected channel *Ch*.

NOTE

When measurement mode is Wide/Narrow, trace 5 to trace 8 are disabled.
When measurement mode is Narrow/Narrow, trace 1 or trace 5 is disabled.

Variable

Parameter	*Value*
Description	Active trace
Data Type	Long integer type (Long)
Range	1 ~ 8
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.TR1.ALLTrace.ACTive = Var
Var = SCPI.CALCulate.TR1.ALLTrace.ACTive
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:ALLTrace:ACTive {1 ~ 8}

:CALCulate:TR[1-1]:ALLTrace:ACTive?

Query Response

{1 ~ 8} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:ALLTrace:ACTive 1 "
20 OUTPUT 717;":CALCulate:TR1:ALLTrace:ACTive ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).ALLTrace.BDMarker.X.COUPle.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).ALLTrace.BDMarker.X.COUPle.STATe = *Value*

Value = SCPI.CALCulate.TR(Ch).ALLTrace.BDMarker.X.COUPle.STATe

Description

This command sets/gets x band marker couple on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	X band marker couple (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.TR1.ALLTrace.BDMarker.X.COUPle.STATe = Var
Var = SCPI.CALCulate.TR1.ALLTrace.BDMarker.X.COUPle.STATe
```

Equivalent Key

TR Menu: **Marker Search** > **Couple**

TR Menu: **Marker Function** > **Couple**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:ALLTrace:BDMarker:X:COUPle:STATe {ON|OFF|1|0}

:CALCulate:TR[1-1]:ALLTrace:BDMarker:X:COUPle:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:ALLTrace:BDMarker:X:COUPle:STATe 1"
20 OUTPUT 717;":CALCulate:TR1:ALLTrace:BDMarker:X:COUPle:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).ALLTrace.LIMit.FAIL

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.TR(Ch).ALLTrace.LIMit.FAIL

Description

This command gets limit test pass/fail, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Result of limit test
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Boolean
Var= 1
Var = SCPI.CALCulate.TR1.ALLTrace.LIMit.FAIL
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:ALLTrace:LIMit:FAIL?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:ALLTrace:LIMit:FAIL ?"
20 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.COUPle.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.COUPle.STATe = *Value*

Value = SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.COUPle.STATe

Description

This command sets/gets marker couple on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker Coupling Function state (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.TR1.ALLTrace.MARKer.COUPle.STATe = Var
Var = SCPI.CALCulate.TR1.ALLTrace.MARKer.COUPle.STATe
```

Related Objects

SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.DISCrete.STATe

SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.REFerence.NUMBer

SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.REFerence.STATe

Equivalent Key

TR Menu: **Marker** > **Couple**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:ALLTrace:MARKer:COUPle:STATe {ON|OFF|1|0}

:CALCulate:TR[1-1]:ALLTrace:MARKer:COUPle:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:ALLTrace:MARKer:COUPle:STATe 1"
20 OUTPUT 717;":CALCulate:TR1:ALLTrace:MARKer:COUPle:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.DISCrete.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.DISCrete.STATe = *Value*

Value = SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.DISCrete.STATe

Description

This command sets/gets marker discrete on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Enables/Disables marker discrete
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.TR1.ALLTrace.MARKer.DISCrete.STATe = Var
Var = SCPI.CALCulate.TR1.ALLTrace.MARKer.DISCrete.STATe
```

Related Objects

SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.COUPle.STATe

SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.REFerence.NUMBer

SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.REFerence.STATe

Equivalent Key

TR Menu: **Marker** > **More Functions** > **Discrete**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:ALLTrace:MARKer:DISCrete:STATe {ON|OFF|1|0}

:CALCulate:TR[1-1]:ALLTrace:MARKer:DISCrete:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:ALLTrace:MARKer:DISCrete:STATe 1"
20 OUTPUT 717;":CALCulate:TR1:ALLTrace:MARKer:DISCrete:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.REFerence.NUMBer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.REFerence.NUMBer = *Value*

Value = SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.REFerence.NUMBer

Description

This command sets/gets reference marker number., for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker reference number
Data Type	Long integer type (Long)
Range	1 ~ 10
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.TR1.ALLTrace.MARKer.REFerence.NUMBer = Var
Var = SCPI.CALCulate.TR1.ALLTrace.MARKer.REFerence.NUMBer
```

Related Objects

SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.COUPle.STATe

SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.DISCrete.STATe

SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.REFerence.STATe

Equivalent Key

TR Menu: **Marker** > **More Functions** > **Ref Marker**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:ALLTrace:MARKer:REFerence:NUMBer {1 ~ 10}

:CALCulate:TR[1-1]:ALLTrace:MARKer:REFerence:NUMBer?

Query Response

{1 ~ 10} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:ALLTrace:MARKer:REFerence:NUMBer 1 "
20 OUTPUT 717;":CALCulate:TR1:ALLTrace:MARKer:REFerence:NUMBer ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.REFerence.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.REFerence.STATe = *Value*

Value = SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.REFerence.STATe

Description

This command sets/gets delta marker mode on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Delta marker mode (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.TR1.ALLTrace.MARKer.REFerence.STATe = Var
Var = SCPI.CALCulate.TR1.ALLTrace.MARKer.REFerence.STATe
```

Related Objects

SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.COUPle.STATe

SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.DISCrete.STATe

SCPI.CALCulate.TR(Ch).ALLTrace.MARKer.REFerence.NUMBer

Equivalent Key

TR Menu: **Marker** > **More Functions** > **Ref Marker Mode**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:ALLTrace:MARKer:REFerence:STATe {ON|OFF|1|0}

:CALCulate:TR[1-1]:ALLTrace:MARKer:REFerence:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:ALLTrace:MARKer:REFerence:STATe 1"
20 OUTPUT 717;":CALCulate:TR1:ALLTrace:MARKer:REFerence:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).NARRow(Nr).DATA.RDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).NARRow(Nr).DATA.RDATa = *Value*

Value = SCPI.CALCulate.TR(Ch).NARRow(Nr).DATA.RDATa

Description

This command sets/gets measurement raw data, for the selected narrow *Nr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Measurement raw data
Data Type	Variant type Array (Range)
Range	1...4692
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 4692) as Variant
SCPI.CALCulate.TR1.NARRow1.DATA.RDATa = Var
Var = SCPI.CALCulate.TR1.NARRow1.DATA.RDATa
```

Related Objects

SCPI.CALCulate.TR(Ch).NARRow(Nr).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:NARRow[1-2]:DATA:RDATa {1...4692}

:CALCulate:TR[1-1]:NARRow[1-2]:DATA:RDATa?

Query Response

{1...4692} <newline><^END>

Example of use

```
10 Dim A(1:4692)
20 OUTPUT 717;":CALCulate:TR1:NARRow1:DATA:RDATa 1,4692"
30 OUTPUT 717;":CALCulate:TR1:NARRow1:DATA:RDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.TR(Ch).NARRow(Nr).DATA.XDATa

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.TR(Ch).NARRow(Nr).DATA.XDATa

Description

This command gets X axis data, for the selected narrow *Nr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	X axis data
Data Type	Variant type Array (Range)
Range	1...1564
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1564) as Variant
Var = SCPI.CALCulate.TR1.NARRow1.DATA.XDATa
```

Related Objects

SCPI.CALCulate.TR(Ch).NARRow(Nr).DATA.RDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:NARRow[1-2]:DATA:XDATa?

Query Response

{1...1564} <newline><^END>

Example of use

```
10 Dim A(1:1564)
20 OUTPUT 717;":CALCulate:TR1:NARRow1:DATA:XDATa ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).ALLMarker.ACTive

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).ALLMarker.ACTive = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).ALLMarker.ACTive

Description

This command sets/gets active marker, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Active marker
Data Type	Long integer type (Long)
Range	1 ~ 10
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.TR1.TRACe4.ALLMarker.ACTive = Var
Var = SCPI.CALCulate.TR1.TRACe4.ALLMarker.ACTive
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:ALLMarker:ACTive {1 ~ 10}

:CALCulate:TR[1-1]:TRACe[1-8]:ALLMarker:ACTive?

Query Response

{1 ~ 10} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:ALLMarker:ACTive 1 "
20 OUTPUT 717;":CALCulate:TR1:TRACe4:ALLMarker:ACTive ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

Description

This command sets/gets marker search X range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search X range source
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	FRANge
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.TR1.TRACe4.ALLMarker.SEARch.DOMain.X = Var
Var = SCPI.CALCulate.TR1.TRACe4.ALLMarker.SEARch.DOMain.X
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

SCPI.CALCulate.TR(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Equivalent Key

TR Menu: **Marker Search** > **Search Range (X)**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:ALLMarker:SEARch:DOMain:X {FRANge|BDMarker}

:CALCulate:TR[1-1]:TRACe[1-8]:ALLMarker:SEARch:DOMain:X?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:ALLMarker:SEARch:DOMain:X FRANge"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:ALLMarker:SEARch:DOMain:X ?"
30 ENTER 717;A$
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Description

This command sets/gets marker search Y range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search Y range
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	FRANge
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.TR1.TRACe4.ALLMarker.SEARch.DOMain.Y = Var
Var = SCPI.CALCulate.TR1.TRACe4.ALLMarker.SEARch.DOMain.Y
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

SCPI.CALCulate.TR(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Equivalent Key

TR Menu: **Marker Search** > **Search Range (Y)**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:ALLMarker:SEARch:DOMain:Y {FRANge|BDMarker}

:CALCulate:TR[1-1]:TRACe[1-8]:ALLMarker:SEARch:DOMain:Y?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:ALLMarker:SEARch:DOMain:Y FRANge"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:ALLMarker:SEARch:DOMain:Y ?"
30 ENTER 717;A$
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Description

This command sets search peak all, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.TR1.TRACe4.ALLMarker.SEARch.PEAK

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

SCPI.CALCulate.TR(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Equivalent Key

TR Menu: **Marker Search** > **Peak** > **Search Peak All**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:ALLMarker:SEARch:PEAK

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:ALLMarker:SEARch:PEAK"
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.CENTer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.CENTer = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.CENTer

Description

This command sets/gets band marker X center, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X center
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.TR1.TRACe4.BDMarker.X.CENTer = Var
Var = SCPI.CALCulate.TR1.TRACe4.BDMarker.X.CENTer
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

TR Menu: **Marker Search** > **Band Marker X** > **Center**

TR Menu: **Marker Function** > **Band Marker X** > **Center**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:BDMarker:X:CENTer {-1T ~ 1T}

:CALCulate:TR[1-1]:TRACe[1-8]:BDMarker:X:CENTer?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:BDMarker:X:CENTer -10000000000"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:BDMarker:X:CENTer ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.SPAN

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.SPAN = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.SPAN

Description

This command sets/gets band marker X span, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X span
Data Type	Double precision floating point type (Double)
Range	0 ~ 2T
Preset Value	2T
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.TR1.TRACe4.BDMarker.X.SPAN = Var
Var = SCPI.CALCulate.TR1.TRACe4.BDMarker.X.SPAN
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

TR Menu: **Marker Search** > **Band Marker X** > **Span**

TR Menu: **Marker Function** > **Band Marker X** > **Span**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:BDMarker:X:SPAN {0 ~ 2T}

:CALCulate:TR[1-1]:TRACe[1-8]:BDMarker:X:SPAN?

Query Response

{0 ~ 2T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:BDMarker:X:SPAN 0"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:BDMarker:X:SPAN ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.STARt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.STARt = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.STARt

Description

This command sets/gets band marker X start, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X start
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	-1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.TR1.TRACe4.BDMarker.X.STARt = Var
Var = SCPI.CALCulate.TR1.TRACe4.BDMarker.X.STARt
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

TR Menu: **Marker Search** > **Band Marker X** > **Start**

TR Menu: **Marker Function** > **Band Marker X** > **Start**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:BDMarker:X:STARt {-1T ~ 1T}

:CALCulate:TR[1-1]:TRACe[1-8]:BDMarker:X:STARt?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:BDMarker:X:STARt -10000000000"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:BDMarker:X:STARt ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.STATe = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.STATe

Description

This command sets/gets band marker visible on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker visibility (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.TR1.TRACe4.BDMarker.X.STATe = Var
Var = SCPI.CALCulate.TR1.TRACe4.BDMarker.X.STATe
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

TR Menu: **Marker Search** > **Band Marker X** > **Band Marker X**

TR Menu: **Marker Function** > **Band Marker X** > **Band Marker X**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:BDMarker:X:STATe {ON|OFF|1|0}

:CALCulate:TR[1-1]:TRACe[1-8]:BDMarker:X:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:BDMarker:X:STATe 1"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:BDMarker:X:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.STOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.STOP = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.STOP

Description

This command sets/gets band marker X stop, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X stop
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.TR1.TRACe4.BDMarker.X.STOP = Var
Var = SCPI.CALCulate.TR1.TRACe4.BDMarker.X.STOP
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.X.STATe

Equivalent Key

TR Menu: **Marker Search** > **Band Marker X** > **Stop**

TR Menu: **Marker Function** > **Band Marker X** > **Stop**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:BDMarker:X:STOP {-1T ~ 1T}

:CALCulate:TR[1-1]:TRACe[1-8]:BDMarker:X:STOP?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:BDMarker:X:STOP -10000000000"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:BDMarker:X:STOP ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.CENTer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.CENTer = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.CENTer

Description

This command sets/gets band marker Y center, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y center
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.TR1.TRACe4.BDMarker.Y.CENTer = Var
Var = SCPI.CALCulate.TR1.TRACe4.BDMarker.Y.CENTer
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

TR Menu: **Marker Search** > **Band Marker Y** > **Center**

TR Menu: **Marker Function** > **Band Marker Y** > **Center**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:BDMarker:Y:CENTer {-1T ~ 1T}

:CALCulate:TR[1-1]:TRACe[1-8]:BDMarker:Y:CENTer?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:BDMarker:Y:CENTer -10000000000"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:BDMarker:Y:CENTer ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.SPAN

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.SPAN = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.SPAN

Description

This command sets/gets band marker Y span, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y span
Data Type	Double precision floating point type (Double)
Range	0 ~ 2T
Preset Value	2T
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.TR1.TRACe4.BDMarker.Y.SPAN = Var
Var = SCPI.CALCulate.TR1.TRACe4.BDMarker.Y.SPAN
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

TR Menu: **Marker Search** > **Band Marker Y** > **Span**

TR Menu: **Marker Function** > **Band Marker Y** > **Span**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:BDMarker:Y:SPAN {0 ~ 2T}

:CALCulate:TR[1-1]:TRACe[1-8]:BDMarker:Y:SPAN?

Query Response

{0 ~ 2T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:BDMarker:Y:SPAN 0"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:BDMarker:Y:SPAN ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.STARt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.STARt = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.STARt

Description

This command sets/gets band marker Y start, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y start
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	-1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.TR1.TRACe4.BDMarker.Y.STARt = Var
Var = SCPI.CALCulate.TR1.TRACe4.BDMarker.Y.STARt
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

TR Menu: **Marker Search** > **Band Marker Y** > **Start**

TR Menu: **Marker Function** > **Band Marker Y** > **Start**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:BDMarker:Y:STARt {-1T ~ 1T}

:CALCulate:TR[1-1]:TRACe[1-8]:BDMarker:Y:STARt?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:BDMarker:Y:STARt -10000000000"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:BDMarker:Y:STARt ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.STATe = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.STATe

Description

This command sets/gets band marker visible on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y state
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.TR1.TRACe4.BDMarker.Y.STATe = Var
Var = SCPI.CALCulate.TR1.TRACe4.BDMarker.Y.STATe
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

TR Menu: **Marker Search** > **Band Marker Y** > **Band Marker Y**

TR Menu: **Marker Function** > **Band Marker Y** > **Band Marker Y**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:BDMarker:Y:STATe {ON|OFF|1|0}

:CALCulate:TR[1-1]:TRACe[1-8]:BDMarker:Y:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:BDMarker:Y:STATe 1"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:BDMarker:Y:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.STOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.STOP = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.STOP

Description

This command sets/gets band marker Y stop, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y stop
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.TR1.TRACe4.BDMarker.Y.STOP = Var
Var = SCPI.CALCulate.TR1.TRACe4.BDMarker.Y.STOP
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.TR(Ch).TRACe(Tr).BDMarker.Y.STATe

Equivalent Key

TR Menu: **Marker Search** > **Band Marker Y** > **Stop**

TR Menu: **Marker Function** > **Band Marker Y** > **Stop**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:BDMarker:Y:STOP {-1T ~ 1T}

:CALCulate:TR[1-1]:TRACe[1-8]:BDMarker:Y:STOP?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:BDMarker:Y:STOP -10000000000"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:BDMarker:Y:STOP ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.COPY

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.COPY = *Value*

Description

This command copies data to user trace, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Copies data to user trace
Data Type	Long integer type (Long)
Range	1 ~ 8
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.TR1.TRACe4.DATA.COPY = Var
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:DATA:COPY {1 ~ 8}

Query Response

{1 ~ 8} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:DATA:COPY 1 "
20 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.FDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.FDATa = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.FDATa

Description

This command sets/gets formatted trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Formatted trace data
Data Type	Variant type Array (Range)
Range	1...1251
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1251) as Variant
SCPI.CALCulate.TR1.TRACe4.DATA.FDATa = Var
Var = SCPI.CALCulate.TR1.TRACe4.DATA.FDATa
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:DATA:FDATa {1...1251}

:CALCulate:TR[1-1]:TRACe[1-8]:DATA:FDATa?

Query Response

{1...1251} <newline><^END>

Example of use

```
10 Dim A(1:1251)
20 OUTPUT 717;":CALCulate:TR1:TRACe4:DATA:FDATa 1,1251"
30 OUTPUT 717;":CALCulate:TR1:TRACe4:DATA:FDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.FMEMory

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.FMEMory = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.FMEMory

Description

This command sets/gets formatted memory data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Formatted memory data
Data Type	Variant type Array (Range)
Range	1...1251
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1251) as Variant
SCPI.CALCulate.TR1.TRACe4.DATA.FMEMory = Var
Var = SCPI.CALCulate.TR1.TRACe4.DATA.FMEMory
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:DATA:FMEMory {1...1251}

:CALCulate:TR[1-1]:TRACe[1-8]:DATA:FMEMory?

Query Response

{1...1251} <newline><^END>

Example of use

```
10 Dim A(1:1251)
20 OUTPUT 717;":CALCulate:TR1:TRACe4:DATA:FMEMory 1,1251"
30 OUTPUT 717;":CALCulate:TR1:TRACe4:DATA:FMEMory ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.UDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.UDATa = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.UDATa

Description

This command sets/gets unformatted trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Unformatted trace data
Data Type	Variant type Array (Range)
Range	1...1251
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1251) as Variant
SCPI.CALCulate.TR1.TRACe4.DATA.UDATa = Var
Var = SCPI.CALCulate.TR1.TRACe4.DATA.UDATa
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:DATA:UDATa {1...1251}

:CALCulate:TR[1-1]:TRACe[1-8]:DATA:UDATa?

Query Response

{1...1251} <newline><^END>

Example of use

```
10 Dim A(1:1251)
20 OUTPUT 717;":CALCulate:TR1:TRACe4:DATA:UDATa 1,1251"
30 OUTPUT 717;":CALCulate:TR1:TRACe4:DATA:UDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.UMEMory

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.UMEMory = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.UMEMory

Description

This command sets/gets unformatted memory data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Unformatted memory data
Data Type	Variant type Array (Range)
Range	1...1251
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1251) as Variant
SCPI.CALCulate.TR1.TRACe4.DATA.UMEMory = Var
Var = SCPI.CALCulate.TR1.TRACe4.DATA.UMEMory
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.TR(Ch).TRACe(Tr).DATA.UDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:DATA:UMEMory {1...1251}

:CALCulate:TR[1-1]:TRACe[1-8]:DATA:UMEMory?

Query Response

{1...1251} <newline><^END>

Example of use

```
10 Dim A(1:1251)
20 OUTPUT 717;":CALCulate:TR1:TRACe4:DATA:UMEMory 1,1251"
30 OUTPUT 717;":CALCulate:TR1:TRACe4:DATA:UMEMory ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).EQUation.TEXT

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).EQUation.TEXT = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).EQUation.TEXT

Description

This command sets/gets equation string, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Equation string
Data Type	Character string type (String)
Range	254 chars
Preset Value	""
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	170, "Invalid equation label" 171, "Invalid equation

	expression"

Examples

```
Dim Var as String
SCPI.CALCulate.TR1.TRACe4.EQUation.TEXT = Var
Var = SCPI.CALCulate.TR1.TRACe4.EQUation.TEXT
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:EQUation:TEXT

:CALCulate:TR[1-1]:TRACe[1-8]:EQUation:TEXT?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:EQUation:TEXT Sample_Text"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:EQUation:TEXT ?"
30 ENTER 717;A$
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.FREQuency

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.FREQuency = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.FREQuency

Description

This command sets/gets frequency format, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Frequency format
Data Type	Character string type (String)
Range	HZ|DHZ|PCT|PPM
Preset Value	HZ
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "HZ"
SCPI.CALCulate.TR1.TRACe4.FORMat.FREQuency = Var
Var = SCPI.CALCulate.TR1.TRACe4.FORMat.FREQuency
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.UNIT

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.WRAP

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.XREFerence

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.XRSPan

Equivalent Key

TR Menu: **Format** > **Frequency Format**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:FORMat:FREQuency {HZ|DHZ|PCT|PPM}

:CALCulate:TR[1-1]:TRACe[1-8]:FORMat:FREQuency?

Query Response

{HZ|DHZ|PCT|PPM} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:FORMat:FREQuency HZ"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:FORMat:FREQuency ?"
30 ENTER 717;A$
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.AUTO

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.AUTO = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.AUTO

Description

This command sets/gets phase reference frequency offset auto calculation on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Phase reference frequency offset auto (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.TR1.TRACe4.FORMat.PHASe.PREFerence.AUTO = Var
Var = SCPI.CALCulate.TR1.TRACe4.FORMat.PHASe.PREFerence.AUTO
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.OFFSet

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.XPOSition

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.XSPan

Equivalent Key

TR Menu: **Setup** > **Recalc Phase Reference** > **Auto Reference**

TR Menu: **Setup** > **Recalc Phase Reference** > **N2 Auto Reference**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:FORMat:PHASe:PREFerence:AUTO {ON|OFF|1|0}

:CALCulate:TR[1-1]:TRACe[1-8]:FORMat:PHASe:PREFerence:AUTO?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:FORMat:PHASe:PREFerence:AUTO 1"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:FORMat:PHASe:PREFerence:AUTO?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.OFFSet

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.OFFSet = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.OFFSet

Description

This command sets/gets phase reference frequency offset, for the selected trace *Tr* of the selected channel *Ch*.

NOTE Disable this function when phase reference frequency offset auto is ON.

Variable

Parameter	*Value*
Description	Phase reference frequency offset
Data Type	Double precision floating point type (Double)
Range	-80M ~ 80M
Preset Value	0
Unit	Hz
Resolution	1m
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
Var= -25600000
SCPI.CALCulate.TR1.TRACe4.FORMat.PHASe.PREFerence.OFFSet = Var
Var = SCPI.CALCulate.TR1.TRACe4.FORMat.PHASe.PREFerence.OFFSet
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.FREQuency

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.UNIT

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.WRAP

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.XREFerence

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.AUTO

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.XPOSition

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.XSPan

Equivalent Key

TR Menu: **Setup** > **Recalc Phase Reference** > **Phase Ref. Offset**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:FORMat:PHASe:PREFerence:OFFSet {-25.6M ~ 25.6M}

:CALCulate:TR[1-1]:TRACe[1-8]:FORMat:PHASe:PREFerence:OFFSet?

Query Response

{-25.6M ~ 25.6M} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:FORMat:PHASe:PREFerence:OFFSet -25600000"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:FORMat:PHASe:PREFerence:OFFSet ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.XPOSition

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.XPOSition = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.XPOSition

Description

This command sets/gets phase reference frequency offset X reference, for the selected trace *Tr* of the selected channel *Ch*.

NOTE This function is disabled when phase reference frequency offset auto is OFF.

Variable

Parameter	*Value*
Description	Phase reference frequency offset X reference
Data Type	Double precision floating point type (Double)
Range	-20 ~ 1.00107936k
Preset Value	1.00107936k
Unit	s
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
Var= -10
SCPI.CALCulate.TR1.TRACe4.FORMat.PHASe.PREFerence.XPOSition = Var
Var = SCPI.CALCulate.TR1.TRACe4.FORMat.PHASe.PREFerence.XPOSition
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.AUTO

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.OFFSet

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.XSPan

Equivalent Key

TR Menu: **Setup** > **Recalc Phase Reference** > **X Reference**

TR Menu: **Setup** > **Recalc Phase Reference** > **N2 X Reference**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:FORMat:PHASe:PREFerence:XPOSition {-20 ~ 1.00107936k}

:CALCulate:TR[1-1]:TRACe[1-8]:FORMat:PHASe:PREFerence:XPOSition?

Query Response

{-20 ~ 1.00107936k} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:FORMat:PHASe:PREFerence:XPOSition -10"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:FORMat:PHASe:PREFerence:XPOSition?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.XSPan

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.XSPan = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.XSPan

Description

This command sets/gets phase reference frequency offset X reference span, for the selected trace *Tr* of the selected channel *Ch*.

NOTE This function is disabled when phase reference frequency offset auto is OFF.

Variable

Parameter	*Value*
Description	Phase reference frequency offset X reference span
Data Type	Double precision floating point type (Double)
Range	0 ~ 1.00007936k
Preset Value	0
Unit	s
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
Var= 100
SCPI.CALCulate.TR1.TRACe4.FORMat.PHASe.PREFerence.XSPan = Var
Var = SCPI.CALCulate.TR1.TRACe4.FORMat.PHASe.PREFerence.XSPan
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.AUTO

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.OFFSet

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.PREFerence.XPOSition

Equivalent Key

TR Menu: **Setup** > **Recalc Phase Reference** > **X Span**

TR Menu: **Setup** > **Recalc Phase Reference** > **N2 X Span**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:FORMat:PHASe:PREFerence:XSPan {0 ~ 1.00007936k}

:CALCulate:TR[1-1]:TRACe[1-8]:FORMat:PHASe:PREFerence:XSPan?

Query Response

{0 ~ 1.00007936k} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:FORMat:PHASe:PREFerence:XSPan 100"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:FORMat:PHASe:PREFerence:XSPan?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.UNIT

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.UNIT = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.UNIT

Description

This command sets/gets TR-phase format, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	TR-phase format
Data Type	Character string type (String)
Range	DEG|RAD|GRAD
Preset Value	DEG
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "DEG"
SCPI.CALCulate.TR1.TRACe4.FORMat.PHASe.UNIT = Var
Var = SCPI.CALCulate.TR1.TRACe4.FORMat.PHASe.UNIT
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.FREQuency

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.WRAP

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.XREFerence

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.XRSPan

Equivalent Key

TR Menu: **Format** > **Phase Unit**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:FORMat:PHASe:UNIT {DEG|RAD|GRAD}

:CALCulate:TR[1-1]:TRACe[1-8]:FORMat:PHASe:UNIT?

Query Response

{DEG|RAD|GRAD} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:FORMat:PHASe:UNIT DEG"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:FORMat:PHASe:UNIT ?"
30 ENTER 717;A$
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.WRAP

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.WRAP = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.WRAP

Description

This command sets/gets phase wrap on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Phase wrap (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.TR1.TRACe4.FORMat.PHASe.WRAP = Var
Var = SCPI.CALCulate.TR1.TRACe4.FORMat.PHASe.WRAP
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.FREQuency

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.UNIT

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.XREFerence

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.XRSPan

Equivalent Key

TR Menu: **Format** > **Wrap Phase**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:FORMat:PHASe:WRAP {ON|OFF|1|0}

:CALCulate:TR[1-1]:TRACe[1-8]:FORMat:PHASe:WRAP?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:FORMat:PHASe:WRAP 1"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:FORMat:PHASe:WRAP ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.XREFerence

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.XREFerence = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.XREFerence

Description

This command sets/gets zero phase X reference, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Zero phase X reference
Data Type	Double precision floating point type (Double)
Range	-20 ~ 1.00107936k
Preset Value	1.00107936k
Unit	s
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.TR1.TRACe4.FORMat.PHASe.XREFerence = Var
Var = SCPI.CALCulate.TR1.TRACe4.FORMat.PHASe.XREFerence
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.FREQuency

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.UNIT

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.WRAP

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.XRSPan

Equivalent Key

TR Menu: **Format** > **Phase X Reference**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:FORMat:PHASe:XREFerence {-20 ~ 1.00107936k}

:CALCulate:TR[1-1]:TRACe[1-8]:FORMat:PHASe:XREFerence?

Query Response

{-20 ~ 1.00107936k} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:FORMat:PHASe:XREFerence 0"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:FORMat:PHASe:XREFerence ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.XRSPan

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.XRSPan = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.XRSPan

Description

This command sets/gets zero phase X reference span, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Zero phase X reference span
Data Type	Double precision floating point type (Double)
Range	0 ~ 1.00007936k
Preset Value	0
Unit	s
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.TR1.TRACe4.FORMat.PHASe.XRSPan = Var
Var = SCPI.CALCulate.TR1.TRACe4.FORMat.PHASe.XRSPan
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.FREQuency

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.UNIT

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.WRAP

SCPI.CALCulate.TR(Ch).TRACe(Tr).FORMat.PHASe.XREFerence

Equivalent Key

TR Menu: **Format** > **X Reference Span**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:FORMat:PHASe:XRSPan {0 ~ 1.00007936k}

:CALCulate:TR[1-1]:TRACe[1-8]:FORMat:PHASe:XRSPan?

Query Response

{0 ~ 1.00007936k} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:FORMat:PHASe:XRSPan 0"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:FORMat:PHASe:XRSPan?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).FUNCtion.DOMain.X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).FUNCtion.DOMain.X = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).FUNCtion.DOMain.X

Description

This command sets/gets X analysis range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	X analysis range
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	BDMarker
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.TR1.TRACe4.FUNCtion.DOMain.X = Var
Var = SCPI.CALCulate.TR1.TRACe4.FUNCtion.DOMain.X
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).FUNCtion.DOMain.Y

Equivalent Key

TR Menu: **Marker Function** > **Analysis Range (X)**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:FUNCtion:DOMain:X {FRANge|BDMarker}

:CALCulate:TR[1-1]:TRACe[1-8]:FUNCtion:DOMain:X?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:FUNCtion:DOMain:X FRANge"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:FUNCtion:DOMain:X ?"
30 ENTER 717;A$
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).FUNCtion.DOMain.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).FUNCtion.DOMain.Y = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).FUNCtion.DOMain.Y

Description

This command sets/gets Y analysis range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Y analysis range
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	BDMarker
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.TR1.TRACe4.FUNCtion.DOMain.Y = Var
Var = SCPI.CALCulate.TR1.TRACe4.FUNCtion.DOMain.Y
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).FUNCtion.DOMain.X

Equivalent Key

TR Menu: **Marker Function** > **Analysis Range (Y)**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:FUNCtion:DOMain:Y {FRANge|BDMarker}

:CALCulate:TR[1-1]:TRACe[1-8]:FUNCtion:DOMain:Y?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:FUNCtion:DOMain:Y FRANge"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:FUNCtion:DOMain:Y ?"
30 ENTER 717;A$
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).FUNCtion.LREGression.DATA_Q a, b

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).FUNCtion.LREGression.DATA_Q a, b

Description

This command gets trace data linear regression, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*a*
Description	Trace data linear regression
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*b*
Description	Trace data linear regression
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
Var = SCPI.CALCulate.TR1.TRACe4.FUNCtion.LREGression.DATA_Q a, b
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).FUNCtion.LREGression.MEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:FUNCtion:LREGression:DATA?

Query Response

{ - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:FUNCtion:LREGression:DATA ?"
30 ENTER 717;A$
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).FUNCtion.LREGression.MEMory_Q a, b

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).FUNCtion.LREGression.MEMory_Q a, b

Description

This command gets memory data linear regression, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*a*
Description	Trace data linear regression
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No

Error	-

Parameter	*b*
Description	Trace data linear regression
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
Var = SCPI.CALCulate.TR1.TRACe4.FUNCtion.LREGression.MEMory_Q a, b
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).FUNCtion.LREGression.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:FUNCtion:LREGression:MEMory?

Query Response

{ - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:FUNCtion:LREGression:MEMory ?"
30 ENTER 717;A$
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak

Description

This command gets trace data statistics, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*mean*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*std_dev*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*peak_to_peak*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-

Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
Var = SCPI.CALCulate.TR1.TRACe4.FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:FUNCtion:STATistics:DATA?

Query Response

{ - - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:FUNCtion:STATistics:DATA ?"
30 ENTER 717;A$
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak

Description

This command gets memory data statistics, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*mean*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*std_dev*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*peak_to_peak*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-

Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
Var = SCPI.CALCulate.TR1.TRACe4.FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).FUNCtion.STATistics.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:FUNCtion:STATistics:MEMory?

Query Response

{ - - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:FUNCtion:STATistics:MEMory ?"
30 ENTER 717;A$
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).FUNCtion.TYPE

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).FUNCtion.TYPE = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).FUNCtion.TYPE

Description

This command sets/gets Analysis type, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Analysis type
Data Type	Character string type (String)
Range	OFF|STATistics|LREGression
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.TR1.TRACe4.FUNCtion.TYPE = Var
Var = SCPI.CALCulate.TR1.TRACe4.FUNCtion.TYPE
```

Equivalent Key

TR Menu: **Marker Function** > **Analysis Type**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:FUNCtion:TYPE {OFF|STATistics|LREGression}

:CALCulate:TR[1-1]:TRACe[1-8]:FUNCtion:TYPE?

Query Response

{OFF|STATistics|LREGression} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:FUNCtion:TYPE OFF"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:FUNCtion:TYPE ?"
30 ENTER 717;A$
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).HOLD

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).HOLD = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).HOLD

Description

This command sets/gets Data hold, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Data hold
Data Type	Character string type (String)
Range	OFF|MAXimum|MINimum
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.TR1.TRACe4.HOLD = Var
Var = SCPI.CALCulate.TR1.TRACe4.HOLD
```

Equivalent Key

TR Menu: **Trace View** > **Data Hold**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:HOLD {OFF|MAXimum|MINimum}

:CALCulate:TR[1-1]:TRACe[1-8]:HOLD?

Query Response

{OFF|MAXimum|MINimum} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:HOLD OFF"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:HOLD ?"
30 ENTER 717;A$
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.FAIL

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.FAIL

Description

This command gets limit test pass/fail, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test result
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Boolean
Var= 1
Var = SCPI.CALCulate.TR1.TRACe4.LIMit.FAIL
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:LIMit:FAIL?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:LIMit:FAIL ?"
20 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.LDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.LDATa = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.LDATa

Description

This command sets/gets expanded lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Lower limit line
Data Type	Variant type Array (Range)
Range	1...1251
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1251) as Variant
SCPI.CALCulate.TR1.TRACe4.LIMit.LOWer.LDATa = Var
Var = SCPI.CALCulate.TR1.TRACe4.LIMit.LOWer.LDATa
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:LIMit:LOWer:LDATa {1...1251}

:CALCulate:TR[1-1]:TRACe[1-8]:LIMit:LOWer:LDATa?

Query Response

{1...1251} <newline><^END>

Example of use

```
10 Dim A(1:1251)
20 OUTPUT 717;":CALCulate:TR1:TRACe4:LIMit:LOWer:LDATa 1,1251"
30 OUTPUT 717;":CALCulate:TR1:TRACe4:LIMit:LOWer:LDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

Description

This command clears lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.TR1.TRACe4.LIMit.LOWer.SEGMent.CLEar

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:LIMit:LOWer:SEGMent:CLEar

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:LIMit:LOWer:SEGMent:CLEar"
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

Description

This command sets/gets number of lower limit segments, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Lower limit segments
Data Type	Long integer type (Long)
Range	1 ~ 100
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.TR1.TRACe4.LIMit.LOWer.SEGMent.COUNt = Var
Var = SCPI.CALCulate.TR1.TRACe4.LIMit.LOWer.SEGMent.COUNt
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:LIMit:LOWer:SEGMent:COUNt {1 ~ 100}

:CALCulate:TR[1-1]:TRACe[1-8]:LIMit:LOWer:SEGMent:COUNt?

Query Response

{1 ~ 100} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:LIMit:LOWer:SEGMent:COUNt 1 "
20 OUTPUT 717;":CALCulate:TR1:TRACe4:LIMit:LOWer:SEGMent:COUNt ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Description

This command sets/gets lower limit line data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Lower limit data
Data Type	Variant type Array (Range)
Range	1...400
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 400) as Variant
SCPI.CALCulate.TR1.TRACe4.LIMit.LOWer.SEGMent.DATA = Var
Var = SCPI.CALCulate.TR1.TRACe4.LIMit.LOWer.SEGMent.DATA
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:LIMit:LOWer:SEGMent:DATA {1...400}

:CALCulate:TR[1-1]:TRACe[1-8]:LIMit:LOWer:SEGMent:DATA?

Query Response

{1...400} <newline><^END>

Example of use

```
10 Dim A(1:400)
20 OUTPUT 717;":CALCulate:TR1:TRACe4:LIMit:LOWer:SEGMent:DATA 1,400"
30 OUTPUT 717;":CALCulate:TR1:TRACe4:LIMit:LOWer:SEGMent:DATA ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.REPort.DATA

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.REPort.DATA

Description

This command gets pass/fail list, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test report
Data Type	Variant type Array (Range)
Range	1...1251
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1251) as Variant
Var = SCPI.CALCulate.TR1.TRACe4.LIMit.REPort.DATA
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:LIMit:REPort[:DATA]?

Query Response

{1...1251} <newline><^END>

Example of use

```
10 Dim A(1:1251)
20 OUTPUT 717;":CALCulate:TR1:TRACe4:LIMit:REPort[:DATA] ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.STATe = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.STATe

Description

This command sets/gets limit test on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.TR1.TRACe4.LIMit.STATe = Var
Var = SCPI.CALCulate.TR1.TRACe4.LIMit.STATe
```

Equivalent Key

TR Menu: **Display** > **Limit Test** > **Limit Test**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:LIMit[:STATe] {ON|OFF|1|0}

:CALCulate:TR[1-1]:TRACe[1-8]:LIMit[:STATe]?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:LIMit[:STATe] 1"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:LIMit[:STATe] ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.LDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.LDATa = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.LDATa

Description

This command sets/gets expanded upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Upper limit line
Data Type	Variant type Array (Range)
Range	1...1251
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1251) as Variant
SCPI.CALCulate.TR1.TRACe4.LIMit.UPPer.LDATa = Var
Var = SCPI.CALCulate.TR1.TRACe4.LIMit.UPPer.LDATa
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:LIMit:UPPer:LDATa {1...1251}

:CALCulate:TR[1-1]:TRACe[1-8]:LIMit:UPPer:LDATa?

Query Response

{1...1251} <newline><^END>

Example of use

```
10 Dim A(1:1251)
20 OUTPUT 717;":CALCulate:TR1:TRACe4:LIMit:UPPer:LDATa 1,1251"
30 OUTPUT 717;":CALCulate:TR1:TRACe4:LIMit:UPPer:LDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

Description

This command clears upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.TR1.TRACe4.LIMit.UPPer.SEGMent.CLEar

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:LIMit:UPPer:SEGMent:CLEar

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:LIMit:UPPer:SEGMent:CLEar"
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Description

This command sets/gets number of upper limit segments, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Upper limit segment
Data Type	Long integer type (Long)
Range	1 ~ 100
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.TR1.TRACe4.LIMit.UPPer.SEGMent.COUNt = Var
Var = SCPI.CALCulate.TR1.TRACe4.LIMit.UPPer.SEGMent.COUNt
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:LIMit:UPPer:SEGMent:COUNt {1 ~ 100}

:CALCulate:TR[1-1]:TRACe[1-8]:LIMit:UPPer:SEGMent:COUNt?

Query Response

{1 ~ 100} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:LIMit:UPPer:SEGMent:COUNt 1 "
20 OUTPUT 717;":CALCulate:TR1:TRACe4:LIMit:UPPer:SEGMent:COUNt ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Description

This command sets/gets upper limit line data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Upper limit line data
Data Type	Variant type Array (Range)
Range	1...400
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 400) as Variant
SCPI.CALCulate.TR1.TRACe4.LIMit.UPPer.SEGMent.DATA = Var
Var = SCPI.CALCulate.TR1.TRACe4.LIMit.UPPer.SEGMent.DATA
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.TR(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:LIMit:UPPer:SEGMent:DATA {1...400}

:CALCulate:TR[1-1]:TRACe[1-8]:LIMit:UPPer:SEGMent:DATA?

Query Response

{1...400} <newline><^END>

Example of use

```
10 Dim A(1:400)
20 OUTPUT 717;":CALCulate:TR1:TRACe4:LIMit:UPPer:SEGMent:DATA 1,400"
30 OUTPUT 717;":CALCulate:TR1:TRACe4:LIMit:UPPer:SEGMent:DATA ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).LINE.A

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).LINE.A = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).LINE.A

Description

This command sets/gets Parameter 'A' as line (Y = AX + B), for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Parameter 'A' in line (Y = AX + B)
Data Type	Double precision floating point type (Double)
Range	-500T ~ 500T
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -5000000000000
SCPI.CALCulate.TR1.TRACe4.LINE.A = Var
Var = SCPI.CALCulate.TR1.TRACe4.LINE.A
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).LINE.A

SCPI.CALCulate.TR(Ch).TRACe(Tr).LINE.B

SCPI.CALCulate.TR(Ch).TRACe(Tr).LINE.MEMory

Equivalent Key

TR Menu: **Trace View** > **Memory Trace** > **Line (Y = AX + B)** > **A**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:LINE:A {-500T ~ 500T}

:CALCulate:TR[1-1]:TRACe[1-8]:LINE:A?

Query Response

{-500T ~ 500T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:LINE:A -5000000000000"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:LINE:A ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).LINE.B

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).LINE.B = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).LINE.B

Description

This command sets/gets Parameter 'B' as line (Y = AX + B), for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Parameter 'B' in line (Y = AX + B)
Data Type	Double precision floating point type (Double)
Range	-500T ~ 500T
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -5000000000000
SCPI.CALCulate.TR1.TRACe4.LINE.B = Var
Var = SCPI.CALCulate.TR1.TRACe4.LINE.B
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).LINE.A

SCPI.CALCulate.TR(Ch).TRACe(Tr).LINE.MEMory

Equivalent Key

TR Menu: **Trace View** > **Memory Trace** > **Line (Y = AX + B)** > **B**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:LINE:B {-500T ~ 500T}

:CALCulate:TR[1-1]:TRACe[1-8]:LINE:B?

Query Response

{-500T ~ 500T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:LINE:B -5000000000000"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:LINE:B ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).LINE.MEMory

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).LINE.MEMory

Description

This command sets line (Y = AX + B) data to Memory trace, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.TR1.TRACe4.LINE.MEMory

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).LINE.A

SCPI.CALCulate.TR(Ch).TRACe(Tr).LINE.B

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:LINE:MEMory

Example of use

10 OUTPUT 717;":CALCulate:TR1:TRACe4:LINE:MEMory"

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

Description

This command sets marker search LPEak, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.TR1.TRACe4.MARKer10.SEARch.EXECute.LPEak

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

TR Menu: **Marker Search** > **Peak** > **Search Left**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:EXECute:LPEak

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:MARKer10:SEARch:EXECute:LPEak"
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

Description

This command sets marker search LTARget, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.TR1.TRACe4.MARKer10.SEARch.EXECute.LTARget

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

TR Menu: **Marker Search** > **Target** > **Search Left**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:EXECute:LTARget

Example of use

10 OUTPUT 717;":CALCulate:TR1:TRACe4:MARKer10:SEARch:EXECute:LTARget"

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

Description

This command sets marker search MAXimum, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.TR1.TRACe4.MARKer10.SEARch.EXECute.MAXimum

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

TR Menu: **Marker Search** > **Search Max**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:EXECute:MAXimum

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:MARKer10:SEARch:EXECute:MAXimum"
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

Description

This command sets marker search MINimum, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.TR1.TRACe4.MARKer10.SEARch.EXECute.MINimum

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

TR Menu: **Marker Search** > **Search Min**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:EXECute:MINimum

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:MARKer10:SEARch:EXECute:MINimum"
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

Description

This command sets marker search PEAK, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.TR1.TRACe4.MARKer10.SEARch.EXECute.PEAK

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

TR Menu: **Marker Search** > **Peak** > **Search Peak**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:EXECute:PEAK

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:MARKer10:SEARch:EXECute:PEAK"
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

Description

This command sets marker search RPEak, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.TR1.TRACe4.MARKer10.SEARch.EXECute.RPEak

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

TR Menu: **Marker Search** > **Peak** > **Search Right**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:EXECute:RPEak

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:MARKer10:SEARch:EXECute:RPEak"
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

Description

This command sets marker search RTARget, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.TR1.TRACe4.MARKer10.SEARch.EXECute.RTARget

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

TR Menu: **Marker Search** > **Target** > **Search Right**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:EXECute:RTARget

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:MARKer10:SEARch:EXECute:RTARget"
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Description

This command sets marker search TARGet, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.TR1.TRACe4.MARKer10.SEARch.EXECute.TARGet

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

Equivalent Key

TR Menu: **Marker Search** > **Target** > **Search Target**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:EXECute:TARGet

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:MARKer10:SEARch:EXECute:TARGet"
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion

Description

This command sets/gets marker-search-peak excursion value, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search peak excursion value
Data Type	Double precision floating point type (Double)
Range	0 ~ 10G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.TR1.TRACe4.MARKer10.SEARch.PEAK.EXCursion = Var
Var = SCPI.CALCulate.TR1.TRACe4.MARKer10.SEARch.PEAK.EXCursion
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity

Equivalent Key

TR Menu: **Marker Search** > **Peak** > **Peak Excursion**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:PEAK:EXCursion {0 ~ 10G}

:CALCulate:TR[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:PEAK:EXCursion?

Query Response

{0 ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:MARKer10:SEARch:PEAK:EXCursion 0"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:MARKer10:SEARch:PEAK:EXCursion ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity

Description

This command sets/gets marker-search-peak polarity type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search peak polarity type
Data Type	Character string type (String)
Range	POSitive|NEGative|BOTH
Preset Value	POSitive
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "POSitive"
SCPI.CALCulate.TR1.TRACe4.MARKer10.SEARch.PEAK.POLarity = Var
Var = SCPI.CALCulate.TR1.TRACe4.MARKer10.SEARch.PEAK.POLarity
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion

Equivalent Key

TR Menu: **Marker Search** > **Peak** > **Peak Polarity**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:PEAK:POLarity {POSitive|NEGative|BOTH}

:CALCulate:TR[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:PEAK:POLarity?

Query Response

{POSitive|NEGative|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:MARKer10:SEARch:PEAK:POLarity POSitive"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:MARKer10:SEARch:PEAK:POLarity ?"
30 ENTER 717;A$
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition

Description

This command sets/gets marker-target transition type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker target transition type
Data Type	Character string type (String)
Range	POSitive|NEGative|BOTH
Preset Value	BOTH
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as String
Var= "POSitive"
SCPI.CALCulate.TR1.TRACe4.MARKer10.SEARch.TARGet.TRANsition = Var
Var = SCPI.CALCulate.TR1.TRACe4.MARKer10.SEARch.TARGet.TRANsition
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y

Equivalent Key

TR Menu: **Marker Search** > **Target** > **Target Transition**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:TARGet:TRANsition {POSitive|NEGative|BOTH}

:CALCulate:TR[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:TARGet:TRANsition?

Query Response

{POSitive|NEGative|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:MARKer10:SEARch:TARGet:TRANsition POSitive"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:MARKer10:SEARch:TARGet:TRANsition ?"
30 ENTER 717;A$
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y

Description

This command sets/gets marker target value, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker target value
Data Type	Double precision floating point type (Double)
Range	-10G ~ 10G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.TR1.TRACe4.MARKer10.SEARch.TARGet.Y = Var
Var = SCPI.CALCulate.TR1.TRACe4.MARKer10.SEARch.TARGet.Y
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition

Equivalent Key

TR Menu: **Marker Search** > **Target** > **Target Value**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:TARGet:Y {-10G ~ 10G}

:CALCulate:TR[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:TARGet:Y?

Query Response

{-10G ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:MARKer10:SEARch:TARGet:Y -10000000000"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:MARKer10:SEARch:TARGet:Y ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE

Description

This command sets/gets search tracking type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Tracking type
Data Type	Character string type (String)
Range	OFF|MAXimum|MINimum|PEAK|TARGet
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.TR1.TRACe4.MARKer10.SEARch.TRACking.TYPE = Var
Var = SCPI.CALCulate.TR1.TRACe4.MARKer10.SEARch.TRACking.TYPE
```

Equivalent Key

TR Menu: **Marker Search** > **Tracking**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:TRACking:TYPE {OFF|MAXimum|MINimum|PEAK|TARGet}

:CALCulate:TR[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:TRACking:TYPE?

Query Response

{OFF|MAXimum|MINimum|PEAK|TARGet} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:MARKer10:SEARch:TRACking:TYPE OFF"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:MARKer10:SEARch:TRACking:TYPE ?"
30 ENTER 717;A$
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).STATe = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).STATe

Description

This command sets/gets marker visible on/off, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker visible (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.TR1.TRACe4.MARKer10.STATe = Var
Var = SCPI.CALCulate.TR1.TRACe4.MARKer10.STATe
```

Related Objects

Equivalent Key

TR Menu: **Marker** > **Clear Marker Menu** > **Marker 1**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:MARKer[1-10]:STATe {ON|OFF|1|0}

:CALCulate:TR[1-1]:TRACe[1-8]:MARKer[1-10]:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:MARKer10:STATe 1"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:MARKer10:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).X = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).X

Description

This command sets/gets marker x position, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker X position
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-50m
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
SCPI.CALCulate.TR1.TRACe4.MARKer10.X = Var
Var = SCPI.CALCulate.TR1.TRACe4.MARKer10.X
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).Y

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:MARKer[1-10]:X

:CALCulate:TR[1-1]:TRACe[1-8]:MARKer[1-10]:X?

Query Response

{Double} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:MARKer10:X 0"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:MARKer10:X ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).Y

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).Y

Description

This command gets marker y position, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker Y position
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var = SCPI.CALCulate.TR1.TRACe4.MARKer10.Y
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).MARKer(Mk).X

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:MARKer[1-10]:Y?

Query Response

{Double} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:MARKer10:Y ?"
20 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).MATH.FUNCtion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).MATH.FUNCtion = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).MATH.FUNCtion

Description

This command sets/gets math operation type, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Math function operation
Data Type	Character string type (String)
Range	NORMal|SUBTract|DIVide|ADD|MULTiply|USER
Preset Value	NORMal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NORMal"
SCPI.CALCulate.TR1.TRACe4.MATH.FUNCtion = Var
Var = SCPI.CALCulate.TR1.TRACe4.MATH.FUNCtion
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).MATH.FUNCtion

SCPI.CALCulate.TR(Ch).TRACe(Tr).MATH.MEMorize

SCPI.CALCulate.TR(Ch).TRACe(Tr).MATH.OFFSet

Equivalent Key

TR Menu: **Trace View** > **Data Math**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:MATH:FUNCtion
{NORMal|SUBTract|DIVide|ADD|MULTiply|USER}

:CALCulate:TR[1-1]:TRACe[1-8]:MATH:FUNCtion?

Query Response

{NORMal|SUBTract|DIVide|ADD|MULTiply|USER} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:MATH:FUNCtion NORMal"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:MATH:FUNCtion ?"
30 ENTER 717;A$
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).MATH.MEMorize

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).MATH.MEMorize

Description

This command copies data to memory, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.TR1.TRACe4.MATH.MEMorize

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).MATH.FUNCtion

SCPI.CALCulate.TR(Ch).TRACe(Tr).MATH.OFFSet

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:MATH:MEMorize

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:MATH:MEMorize"
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).MATH.OFFSet

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).MATH.OFFSet = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).MATH.OFFSet

Description

This command sets/gets Trace Offset, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace Offset
Data Type	Double precision floating point type (Double)
Range	-500G ~ 500G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -500000000000
SCPI.CALCulate.TR1.TRACe4.MATH.OFFSet = Var
Var = SCPI.CALCulate.TR1.TRACe4.MATH.OFFSet
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).MATH.FUNCtion

SCPI.CALCulate.TR(Ch).TRACe(Tr).MATH.MEMorize

Equivalent Key

TR Menu: **Trace View** > **Offset**

TR Menu: **Trace View** > **Marker** > **-Offset**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:MATH:OFFSet {-500G ~ 500G}

:CALCulate:TR[1-1]:TRACe[1-8]:MATH:OFFSet?

Query Response

{-500G ~ 500G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:MATH:OFFSet -500000000000"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:MATH:OFFSet ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).PARameter

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).PARameter

Description

This command gets TR measurement type. Init value changes by tr #., for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	TR measurement type
Data Type	Character string type (String)
Range	FREQuency|POWer|PHASe
Preset Value	FREQuency
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as String
Var= "FREQuency"
Var = SCPI.CALCulate.TR1.TRACe4.PARameter
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:PARameter?

Query Response

{FREQuency|POWer|PHASe} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:PARameter ?"
20 ENTER 717;A$
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).REFerence.FREQuency

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).REFerence.FREQuency = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).REFerence.FREQuency

Description

This command sets/gets frequency reference for DHZ, PCT, PPM, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Frequency reference for DHZ, PCT & PPM
Data Type	Double precision floating point type (Double)
Range	-500G ~ 500G
Preset Value	0
Unit	Hz
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -500000000000
SCPI.CALCulate.TR1.TRACe4.REFerence.FREQuency = Var
Var = SCPI.CALCulate.TR1.TRACe4.REFerence.FREQuency
```

Equivalent Key

TR Menu: **Format** > **Frequency Reference**

TR Menu: **Format** > **Marker** > **Frequency Reference**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:REFerence:FREQuency {-500G ~ 500G}

:CALCulate:TR[1-1]:TRACe[1-8]:REFerence:FREQuency?

Query Response

{-500G ~ 500G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:REFerence:FREQuency -500000000000"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:REFerence:FREQuency ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).SMOothing.APERture

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).SMOothing.APERture = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).SMOothing.APERture

Description

This command sets/gets Smoothing aperture, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Smoothing aperture
Data Type	Double precision floating point type (Double)
Range	50m ~ 25
Preset Value	1.5
Unit	%
Resolution	10m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0.05
SCPI.CALCulate.TR1.TRACe4.SMOothing.APERture = Var
Var = SCPI.CALCulate.TR1.TRACe4.SMOothing.APERture
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).SMOothing.STATe

Equivalent Key

TR Menu: **Trace View** > **Aperture**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:SMOothing:APERture {50m ~ 25}

:CALCulate:TR[1-1]:TRACe[1-8]:SMOothing:APERture?

Query Response

{50m ~ 25} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:SMOothing:APERture 0.05"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:SMOothing:APERture ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).TRACe(Tr).SMOothing.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).TRACe(Tr).SMOothing.STATe = *Value*

Value = SCPI.CALCulate.TR(Ch).TRACe(Tr).SMOothing.STATe

Description

This command sets/gets Smoothing on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Smoothing (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.TR1.TRACe4.SMOothing.STATe = Var
Var = SCPI.CALCulate.TR1.TRACe4.SMOothing.STATe
```

Related Objects

SCPI.CALCulate.TR(Ch).TRACe(Tr).SMOothing.APERture

Equivalent Key

TR Menu: **Trace View** > **Smoothing**

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:TRACe[1-8]:SMOothing:STATe {ON|OFF|1|0}

:CALCulate:TR[1-1]:TRACe[1-8]:SMOothing:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:TR1:TRACe4:SMOothing:STATe 1"
20 OUTPUT 717;":CALCulate:TR1:TRACe4:SMOothing:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.TR(Ch).WIDE.DATA.RDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.TR(Ch).WIDE.DATA.RDATa = *Value*

Value = SCPI.CALCulate.TR(Ch).WIDE.DATA.RDATa

Description

This command sets/gets measurement raw data, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Measurement raw data
Data Type	Variant type Array (Range)
Range	1...1251
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1251) as Variant
SCPI.CALCulate.TR1.WIDE.DATA.RDATa = Var
Var = SCPI.CALCulate.TR1.WIDE.DATA.RDATa
```

Related Objects

SCPI.CALCulate.TR(Ch).WIDE.DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:WIDE:DATA:RDATa {1...1251}

:CALCulate:TR[1-1]:WIDE:DATA:RDATa?

Query Response

{1...1251} <newline><^END>

Example of use

```
10 Dim A(1:1251)
20 OUTPUT 717;":CALCulate:TR1:WIDE:DATA:RDATa 1,1251"
30 OUTPUT 717;":CALCulate:TR1:WIDE:DATA:RDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.TR(Ch).WIDE.DATA.XDATa

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.TR(Ch).WIDE.DATA.XDATa

Description

This command gets X axis data, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	X axis data
Data Type	Variant type Array (Range)
Range	1...1251
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1251) as Variant
Var = SCPI.CALCulate.TR1.WIDE.DATA.XDATa
```

Related Objects

SCPI.CALCulate.TR(Ch).WIDE.DATA.RDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:TR[1-1]:WIDE:DATA:XDATa?

Query Response

{1...1251} <newline><^END>

Example of use

```
10 Dim A(1:1251)
20 OUTPUT 717;":CALCulate:TR1:WIDE:DATA:XDATa ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.USER(Ch).ALLTrace.ACTive

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).ALLTrace.ACTive = *Value*

Value = SCPI.CALCulate.USER(Ch).ALLTrace.ACTive

Description

This command sets/gets active trace, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Active trace
Data Type	Long integer type (Long)
Range	1 ~ 8
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.USER1.ALLTrace.ACTive = Var
Var = SCPI.CALCulate.USER1.ALLTrace.ACTive
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:ALLTrace:ACTive {1 ~ 8}

:CALCulate:USER[1-1]:ALLTrace:ACTive?

Query Response

{1 ~ 8} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:ALLTrace:ACTive 1 "
20 OUTPUT 717;":CALCulate:USER1:ALLTrace:ACTive ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).ALLTrace.BDMarker.X.COUPle.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).ALLTrace.BDMarker.X.COUPle.STATe = *Value*

Value = SCPI.CALCulate.USER(Ch).ALLTrace.BDMarker.X.COUPle.STATe

Description

This command sets/gets x band marker couple on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	X band marker couple (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.USER1.ALLTrace.BDMarker.X.COUPle.STATe = Var
Var = SCPI.CALCulate.USER1.ALLTrace.BDMarker.X.COUPle.STATe
```

Related Objects

Equivalent Key

USER Menu: **Marker Search** > **Couple**

USER Menu: **Marker Function** > **Couple**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:ALLTrace:BDMarker:X:COUPle:STATe {ON|OFF|1|0}

:CALCulate:USER[1-1]:ALLTrace:BDMarker:X:COUPle:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:ALLTrace:BDMarker:X:COUPle:STATe 1"
20 OUTPUT 717;":CALCulate:USER1:ALLTrace:BDMarker:X:COUPle:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).ALLTrace.LIMit.FAIL

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.USER(Ch).ALLTrace.LIMit.FAIL

Description

This command gets limit test pass/fail, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Result of limit test
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Boolean
Var= 1
Var = SCPI.CALCulate.USER1.ALLTrace.LIMit.FAIL
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:ALLTrace:LIMit:FAIL?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:ALLTrace:LIMit:FAIL ?"
20 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.COUPle.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.COUPle.STATe = *Value*

Value = SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.COUPle.STATe

Description

This command sets/gets marker couple on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker Coupling Function state (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.USER1.ALLTrace.MARKer.COUPle.STATe = Var
Var = SCPI.CALCulate.USER1.ALLTrace.MARKer.COUPle.STATe
```

Related Objects

SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.DISCrete.STATe

SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.REFerence.NUMBer

SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.REFerence.STATe

Equivalent Key

USER Menu: **Marker** > **Couple**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:ALLTrace:MARKer:COUPle:STATe {ON|OFF|1|0}

:CALCulate:USER[1-1]:ALLTrace:MARKer:COUPle:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:ALLTrace:MARKer:COUPle:STATe 1"
20 OUTPUT 717;":CALCulate:USER1:ALLTrace:MARKer:COUPle:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.DISCrete.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.DISCrete.STATe = *Value*

Value = SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.DISCrete.STATe

Description

This command sets/gets marker discrete on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Enables/Disables marker discrete
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.USER1.ALLTrace.MARKer.DISCrete.STATe = Var
Var = SCPI.CALCulate.USER1.ALLTrace.MARKer.DISCrete.STATe
```

Related Objects

SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.COUPle.STATe

SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.REFerence.NUMBer

SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.REFerence.STATe

Equivalent Key

USER Menu: **Marker** > **More Functions** > **Discrete**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:ALLTrace:MARKer:DISCrete:STATe {ON|OFF|1|0}

:CALCulate:USER[1-1]:ALLTrace:MARKer:DISCrete:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:ALLTrace:MARKer:DISCrete:STATe 1"
20 OUTPUT 717;":CALCulate:USER1:ALLTrace:MARKer:DISCrete:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.REFerence.NUMBer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.REFerence.NUMBer = *Value*

Value = SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.REFerence.NUMBer

Description

This command sets/gets reference marker number., for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker reference number
Data Type	Long integer type (Long)
Range	1 ~ 10
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.USER1.ALLTrace.MARKer.REFerence.NUMBer = Var
Var = SCPI.CALCulate.USER1.ALLTrace.MARKer.REFerence.NUMBer
```

Related Objects

SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.COUPle.STATe

SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.DISCrete.STATe

SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.REFerence.STATe

Equivalent Key

USER Menu: **Marker** > **More Functions** > **Ref Marker**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:ALLTrace:MARKer:REFerence:NUMBer {1 ~ 10}

:CALCulate:USER[1-1]:ALLTrace:MARKer:REFerence:NUMBer?

Query Response

{1 ~ 10} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:ALLTrace:MARKer:REFerence:NUMBer 1 "
20 OUTPUT 717;":CALCulate:USER1:ALLTrace:MARKer:REFerence:NUMBer ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.REFerence.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.REFerence.STATe = *Value*

Value = SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.REFerence.STATe

Description

This command sets/gets delta marker mode on/off, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Delta marker mode (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.USER1.ALLTrace.MARKer.REFerence.STATe = Var
Var = SCPI.CALCulate.USER1.ALLTrace.MARKer.REFerence.STATe
```

Related Objects

SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.COUPle.STATe

SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.DISCrete.STATe

SCPI.CALCulate.USER(Ch).ALLTrace.MARKer.REFerence.NUMBer

Equivalent Key

USER Menu: **Marker** > **More Functions** > **Ref Marker Mode**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:ALLTrace:MARKer:REFerence:STATe {ON|OFF|1|0}

:CALCulate:USER[1-1]:ALLTrace:MARKer:REFerence:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:ALLTrace:MARKer:REFerence:STATe 1"
20 OUTPUT 717;":CALCulate:USER1:ALLTrace:MARKer:REFerence:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).ALLMarker.ACTive

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).ALLMarker.ACTive = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).ALLMarker.ACTive

Description

This command sets/gets active marker, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Active marker
Data Type	Long integer type (Long)
Range	1 ~ 10
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.USER1.TRACe8.ALLMarker.ACTive = Var
Var = SCPI.CALCulate.USER1.TRACe8.ALLMarker.ACTive
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:ALLMarker:ACTive {1 ~ 10}

:CALCulate:USER[1-1]:TRACe[1-8]:ALLMarker:ACTive?

Query Response

{1 ~ 10} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:ALLMarker:ACTive 1 "
20 OUTPUT 717;":CALCulate:USER1:TRACe8:ALLMarker:ACTive ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

Description

This command sets/gets marker search X range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search X range source
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	FRANge
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.USER1.TRACe8.ALLMarker.SEARch.DOMain.X = Var
Var = SCPI.CALCulate.USER1.TRACe8.ALLMarker.SEARch.DOMain.X
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

SCPI.CALCulate.USER(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Equivalent Key

USER Menu: **Marker Search** > **Search Range (X)**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:ALLMarker:SEARch:DOMain:X {FRANge|BDMarker}

:CALCulate:USER[1-1]:TRACe[1-8]:ALLMarker:SEARch:DOMain:X?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:ALLMarker:SEARch:DOMain:X FRANge"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:ALLMarker:SEARch:DOMain:X ?"
30 ENTER 717;A$
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Description

This command sets/gets marker search Y range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search Y range
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	FRANge
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.USER1.TRACe8.ALLMarker.SEARch.DOMain.Y = Var
Var = SCPI.CALCulate.USER1.TRACe8.ALLMarker.SEARch.DOMain.Y
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

SCPI.CALCulate.USER(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Equivalent Key

USER Menu: **Marker Search** > **Search Range (Y)**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:ALLMarker:SEARch:DOMain:Y {FRANge|BDMarker}

:CALCulate:USER[1-1]:TRACe[1-8]:ALLMarker:SEARch:DOMain:Y?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:ALLMarker:SEARch:DOMain:Y FRANge"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:ALLMarker:SEARch:DOMain:Y ?"
30 ENTER 717;A$
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).ALLMarker.SEARch.PEAK

Description

This command sets search peak all, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.USER1.TRACe8.ALLMarker.SEARch.PEAK

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.X

SCPI.CALCulate.USER(Ch).TRACe(Tr).ALLMarker.SEARch.DOMain.Y

Equivalent Key

USER Menu: **Marker Search** > **Peak** > **Search Peak All**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:ALLMarker:SEARch:PEAK

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:ALLMarker:SEARch:PEAK"
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.CENTer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.CENTer = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.CENTer

Description

This command sets/gets band marker X center, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X center
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.USER1.TRACe8.BDMarker.X.CENTer = Var
Var = SCPI.CALCulate.USER1.TRACe8.BDMarker.X.CENTer
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

USER Menu: **Marker Search** > **Band Marker X** > **Center**

USER Menu: **Marker Function** > **Band Marker X** > **Center**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:BDMarker:X:CENTer {-1T ~ 1T}

:CALCulate:USER[1-1]:TRACe[1-8]:BDMarker:X:CENTer?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:BDMarker:X:CENTer -10000000000"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:BDMarker:X:CENTer ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.SPAN

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.SPAN = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.SPAN

Description

This command sets/gets band marker X span, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X span
Data Type	Double precision floating point type (Double)
Range	0 ~ 2T
Preset Value	2T
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.USER1.TRACe8.BDMarker.X.SPAN = Var
Var = SCPI.CALCulate.USER1.TRACe8.BDMarker.X.SPAN
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

USER Menu: **Marker Search** > **Band Marker X** > **Span**

USER Menu: **Marker Function** > **Band Marker X** > **Span**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:BDMarker:X:SPAN {0 ~ 2T}

:CALCulate:USER[1-1]:TRACe[1-8]:BDMarker:X:SPAN?

Query Response

{0 ~ 2T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:BDMarker:X:SPAN 0"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:BDMarker:X:SPAN ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.STARt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.STARt = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.STARt

Description

This command sets/gets band marker X start, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X start
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	-1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.USER1.TRACe8.BDMarker.X.STARt = Var
Var = SCPI.CALCulate.USER1.TRACe8.BDMarker.X.STARt
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.STATe

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

USER Menu: **Marker Search** > **Band Marker X** > **Start**

USER Menu: **Marker Function** > **Band Marker X** > **Start**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:BDMarker:X:STARt {-1T ~ 1T}

:CALCulate:USER[1-1]:TRACe[1-8]:BDMarker:X:STARt?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:BDMarker:X:STARt -10000000000"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:BDMarker:X:STARt ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.STATe = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.STATe

Description

This command sets/gets band marker visible on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker visibility (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.USER1.TRACe8.BDMarker.X.STATe = Var
Var = SCPI.CALCulate.USER1.TRACe8.BDMarker.X.STATe
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.STOP

Equivalent Key

USER Menu: **Marker Search** > **Band Marker X** > **Band Marker X**

USER Menu: **Marker Function** > **Band Marker X** > **Band Marker X**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:BDMarker:X:STATe {ON|OFF|1|0}

:CALCulate:USER[1-1]:TRACe[1-8]:BDMarker:X:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:BDMarker:X:STATe 1"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:BDMarker:X:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.STOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.STOP = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.STOP

Description

This command sets/gets band marker X stop, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker X stop
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.USER1.TRACe8.BDMarker.X.STOP = Var
Var = SCPI.CALCulate.USER1.TRACe8.BDMarker.X.STOP
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.CENTer

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.SPAN

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.STARt

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.X.STATe

Equivalent Key

USER Menu: **Marker Search** > **Band Marker X** > **Stop**

USER Menu: **Marker Function** > **Band Marker X** > **Stop**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:BDMarker:X:STOP {-1T ~ 1T}

:CALCulate:USER[1-1]:TRACe[1-8]:BDMarker:X:STOP?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:BDMarker:X:STOP -10000000000"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:BDMarker:X:STOP ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.CENTer

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.CENTer = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.CENTer

Description

This command sets/gets band marker Y center, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y center
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.USER1.TRACe8.BDMarker.Y.CENTer = Var
Var = SCPI.CALCulate.USER1.TRACe8.BDMarker.Y.CENTer
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

USER Menu: **Marker Search** > **Band Marker Y** > **Center**

USER Menu: **Marker Function** > **Band Marker Y** > **Center**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:BDMarker:Y:CENTer {-1T ~ 1T}

:CALCulate:USER[1-1]:TRACe[1-8]:BDMarker:Y:CENTer?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:BDMarker:Y:CENTer -10000000000"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:BDMarker:Y:CENTer ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.SPAN

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.SPAN = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.SPAN

Description

This command sets/gets band marker Y span, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y span
Data Type	Double precision floating point type (Double)
Range	0 ~ 2T
Preset Value	2T
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.USER1.TRACe8.BDMarker.Y.SPAN = Var
Var = SCPI.CALCulate.USER1.TRACe8.BDMarker.Y.SPAN
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

USER Menu: **Marker Search** > **Band Marker Y** > **Span**

USER Menu: **Marker Function** > **Band Marker Y** > **Span**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:BDMarker:Y:SPAN {0 ~ 2T}

:CALCulate:USER[1-1]:TRACe[1-8]:BDMarker:Y:SPAN?

Query Response

{0 ~ 2T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:BDMarker:Y:SPAN 0"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:BDMarker:Y:SPAN ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.STARt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.STARt = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.STARt

Description

This command sets/gets band marker Y start, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y start
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	-1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.USER1.TRACe8.BDMarker.Y.STARt = Var
Var = SCPI.CALCulate.USER1.TRACe8.BDMarker.Y.STARt
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.STATe

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

USER Menu: **Marker Search** > **Band Marker Y** > **Start**

USER Menu: **Marker Function** > **Band Marker Y** > **Start**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:BDMarker:Y:STARt {-1T ~ 1T}

:CALCulate:USER[1-1]:TRACe[1-8]:BDMarker:Y:STARt?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:BDMarker:Y:STARt -10000000000"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:BDMarker:Y:STARt ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.STATe = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.STATe

Description

This command sets/gets band marker visible on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y state
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.USER1.TRACe8.BDMarker.Y.STATe = Var
Var = SCPI.CALCulate.USER1.TRACe8.BDMarker.Y.STATe
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.STOP

Equivalent Key

USER Menu: **Marker Search** > **Band Marker Y** > **Band Marker Y**

USER Menu: **Marker Function** > **Band Marker Y** > **Band Marker Y**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:BDMarker:Y:STATe {ON|OFF|1|0}

:CALCulate:USER[1-1]:TRACe[1-8]:BDMarker:Y:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:BDMarker:Y:STATe 1"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:BDMarker:Y:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.STOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.STOP = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.STOP

Description

This command sets/gets band marker Y stop, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Band marker Y stop
Data Type	Double precision floating point type (Double)
Range	-1T ~ 1T
Preset Value	1T
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.USER1.TRACe8.BDMarker.Y.STOP = Var
Var = SCPI.CALCulate.USER1.TRACe8.BDMarker.Y.STOP
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.CENTer

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.SPAN

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.STARt

SCPI.CALCulate.USER(Ch).TRACe(Tr).BDMarker.Y.STATe

Equivalent Key

USER Menu: **Marker Search** > **Band Marker Y** > **Stop**

USER Menu: **Marker Function** > **Band Marker Y** > **Stop**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:BDMarker:Y:STOP {-1T ~ 1T}

:CALCulate:USER[1-1]:TRACe[1-8]:BDMarker:Y:STOP?

Query Response

{-1T ~ 1T} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:BDMarker:Y:STOP -10000000000"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:BDMarker:Y:STOP ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.COPY

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.COPY = *Value*

Description

This command copies data to user trace, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Copies data to user trace
Data Type	Long integer type (Long)
Range	1 ~ 8
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.USER1.TRACe8.DATA.COPY = Var
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.POINts

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.RDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STARt

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STOP

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UMEMory

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:DATA:COPY {1 ~ 8}

Query Response

{1 ~ 8} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:DATA:COPY 1 "
20 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FDATa = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FDATa

Description

This command sets/gets formatted trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Formatted trace data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.USER1.TRACe8.DATA.FDATa = Var
Var = SCPI.CALCulate.USER1.TRACe8.DATA.FDATa
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.POINts

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.RDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STARt

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STOP

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UMEMory

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:DATA:FDATa {1...1601}

:CALCulate:USER[1-1]:TRACe[1-8]:DATA:FDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:USER1:TRACe8:DATA:FDATa 1,1601"
30 OUTPUT 717;":CALCulate:USER1:TRACe8:DATA:FDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FMEMory

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FMEMory = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FMEMory

Description

This command sets/gets formatted memory data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Formatted memory data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.USER1.TRACe8.DATA.FMEMory = Var
Var = SCPI.CALCulate.USER1.TRACe8.DATA.FMEMory
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.POINts

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.RDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STARt

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STOP

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UMEMory

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:DATA:FMEMory {1...1601}

:CALCulate:USER[1-1]:TRACe[1-8]:DATA:FMEMory?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:USER1:TRACe8:DATA:FMEMory 1,1601"
30 OUTPUT 717;":CALCulate:USER1:TRACe8:DATA:FMEMory ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.POINts

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.POINts

Description

This command gets number of points, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	dBC data
Data Type	Long integer type (Long)
Range	2 ~ 1601
Preset Value	11
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 2
Var = SCPI.CALCulate.USER1.TRACe8.DATA.POINts
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.RDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STARt

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STOP

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UMEMory

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:DATA:POINts?

Query Response

{2 ~ 1601} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:DATA:POINts ?"
20 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.RDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.RDATa = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.RDATa

Description

This command sets/gets raw data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Raw data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.USER1.TRACe8.DATA.RDATa = Var
Var = SCPI.CALCulate.USER1.TRACe8.DATA.RDATa
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.POINts

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STARt

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STOP

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UMEMory

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:DATA:RDATa {1...1601}

:CALCulate:USER[1-1]:TRACe[1-8]:DATA:RDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:USER1:TRACe8:DATA:RDATa 1,1601"
30 OUTPUT 717;":CALCulate:USER1:TRACe8:DATA:RDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STARt

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STARt

Description

This command gets user start x, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	User start x
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var = SCPI.CALCulate.USER1.TRACe8.DATA.STARt
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.POINts

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.RDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STOP

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UMEMory

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:DATA:STARt?

Query Response

{Double} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:DATA:STARt ?"
20 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STOP

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STOP

Description

This command gets user stop x, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	User stop x
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	100
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var = SCPI.CALCulate.USER1.TRACe8.DATA.STOP
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.POINts

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.RDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STARt

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UMEMory

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:DATA:STOP?

Query Response

{Double} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:DATA:STOP ?"
20 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UDATa = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UDATa

Description

This command sets/gets unformatted trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Unformatted trace data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.USER1.TRACe8.DATA.UDATa = Var
Var = SCPI.CALCulate.USER1.TRACe8.DATA.UDATa
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.POINts

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.RDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STARt

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STOP

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UMEMory

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:DATA:UDATa {1...1601}

:CALCulate:USER[1-1]:TRACe[1-8]:DATA:UDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:USER1:TRACe8:DATA:UDATa 1,1601"
30 OUTPUT 717;":CALCulate:USER1:TRACe8:DATA:UDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UMEMory

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UMEMory = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UMEMory

Description

This command sets/gets unformatted memory data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Unformatted memory data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.USER1.TRACe8.DATA.UMEMory = Var
Var = SCPI.CALCulate.USER1.TRACe8.DATA.UMEMory
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.POINts

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.RDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STARt

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STOP

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.XDATa

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:DATA:UMEMory {1...1601}

:CALCulate:USER[1-1]:TRACe[1-8]:DATA:UMEMory?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:USER1:TRACe8:DATA:UMEMory 1,1601"
30 OUTPUT 717;":CALCulate:USER1:TRACe8:DATA:UMEMory ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.XDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.XDATa = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.XDATa

Description

This command sets/gets X axis data and number of points. must be incremental order. NOP >=2., for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	X axis data and number of points
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-224, "Illegal parameter value"

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.USER1.TRACe8.DATA.XDATa = Var
Var = SCPI.CALCulate.USER1.TRACe8.DATA.XDATa
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.COPY

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.FMEMory

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.POINts

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.RDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STARt

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STOP

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.UMEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:DATA:XDATa {1...1601}

:CALCulate:USER[1-1]:TRACe[1-8]:DATA:XDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:USER1:TRACe8:DATA:XDATa 1,1601"
30 OUTPUT 717;":CALCulate:USER1:TRACe8:DATA:XDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).EQUation.TEXT

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).EQUation.TEXT = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).EQUation.TEXT

Description

This command sets/gets equation string, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Equation string
Data Type	Character string type (String)
Range	254 chars
Preset Value	""
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	170, "Invalid equation label" 171, "Invalid equation

	expression"

Examples

```
Dim Var as String
SCPI.CALCulate.USER1.TRACe8.EQUation.TEXT = Var
Var = SCPI.CALCulate.USER1.TRACe8.EQUation.TEXT
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:EQUation:TEXT

:CALCulate:USER[1-1]:TRACe[1-8]:EQUation:TEXT?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:EQUation:TEXT Sample_Text"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:EQUation:TEXT ?"
30 ENTER 717;A$
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).FUNCtion.DOMain.X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).FUNCtion.DOMain.X = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).FUNCtion.DOMain.X

Description

This command sets/gets X analysis range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	X analysis range
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	BDMarker
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.USER1.TRACe8.FUNCtion.DOMain.X = Var
Var = SCPI.CALCulate.USER1.TRACe8.FUNCtion.DOMain.X
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).FUNCtion.DOMain.Y

Equivalent Key

USER Menu: **Marker Function** > **Analysis Range (X)**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:FUNCtion:DOMain:X {FRANge|BDMarker}

:CALCulate:USER[1-1]:TRACe[1-8]:FUNCtion:DOMain:X?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:FUNCtion:DOMain:X FRANge"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:FUNCtion:DOMain:X ?"
30 ENTER 717;A$
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).FUNCtion.DOMain.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).FUNCtion.DOMain.Y = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).FUNCtion.DOMain.Y

Description

This command sets/gets Y analysis range source, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Y analysis range
Data Type	Character string type (String)
Range	FRANge|BDMarker
Preset Value	BDMarker
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FRANge"
SCPI.CALCulate.USER1.TRACe8.FUNCtion.DOMain.Y = Var
Var = SCPI.CALCulate.USER1.TRACe8.FUNCtion.DOMain.Y
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).FUNCtion.DOMain.X

Equivalent Key

USER Menu: **Marker Function** > **Analysis Range (Y)**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:FUNCtion:DOMain:Y {FRANge|BDMarker}

:CALCulate:USER[1-1]:TRACe[1-8]:FUNCtion:DOMain:Y?

Query Response

{FRANge|BDMarker} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:FUNCtion:DOMain:Y FRANge"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:FUNCtion:DOMain:Y ?"
30 ENTER 717;A$
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak

Description

This command gets trace data statistics, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*mean*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*std_dev*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*peak_to_peak*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-

Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
Var = SCPI.CALCulate.USER1.TRACe8.FUNCtion.STATistics.DATA_Q mean, std_dev, peak_to_peak
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:FUNCtion:STATistics:DATA?

Query Response

{ - - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:FUNCtion:STATistics:DATA ?"
30 ENTER 717;A$
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak

Description

This command gets memory data statistics, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*mean*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*std_dev*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*peak_to_peak*
Description	Trace data statistics
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-

1820

Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
Var = SCPI.CALCulate.USER1.TRACe8.FUNCtion.STATistics.MEMory_Q mean, std_dev, peak_to_peak
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).FUNCtion.STATistics.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:FUNCtion:STATistics:MEMory?

Query Response

{ - - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:FUNCtion:STATistics:MEMory ?"
30 ENTER 717;A$
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).FUNCtion.TYPE

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).FUNCtion.TYPE = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).FUNCtion.TYPE

Description

This command sets/gets Analysis type, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Analysis type
Data Type	Character string type (String)
Range	OFF|STATistics
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.USER1.TRACe8.FUNCtion.TYPE = Var
Var = SCPI.CALCulate.USER1.TRACe8.FUNCtion.TYPE
```

Equivalent Key

USER Menu: **Marker Function** > **Analysis Type**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:FUNCtion:TYPE {OFF|STATistics}

:CALCulate:USER[1-1]:TRACe[1-8]:FUNCtion:TYPE?

Query Response

{OFF|STATistics} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:FUNCtion:TYPE OFF"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:FUNCtion:TYPE ?"
30 ENTER 717;A$
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).HOLD

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).HOLD = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).HOLD

Description

This command sets/gets Data hold, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Data hold
Data Type	Character string type (String)
Range	OFF|MAXimum|MINimum
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.USER1.TRACe8.HOLD = Var
Var = SCPI.CALCulate.USER1.TRACe8.HOLD
```

Equivalent Key

USER Menu: **Trace View** > **Data Hold**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:HOLD {OFF|MAXimum|MINimum}

:CALCulate:USER[1-1]:TRACe[1-8]:HOLD?

Query Response

{OFF|MAXimum|MINimum} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:HOLD OFF"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:HOLD ?"
30 ENTER 717;A$
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.FAIL

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.FAIL

Description

This command gets limit test pass/fail, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test result
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Boolean
Var= 1
Var = SCPI.CALCulate.USER1.TRACe8.LIMit.FAIL
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:LIMit:FAIL?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:LIMit:FAIL ?"
20 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.LDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.LDATa = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.LDATa

Description

This command sets/gets expanded lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Lower limit line
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.USER1.TRACe8.LIMit.LOWer.LDATa = Var
Var = SCPI.CALCulate.USER1.TRACe8.LIMit.LOWer.LDATa
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:LIMit:LOWer:LDATa {1...1601}

:CALCulate:USER[1-1]:TRACe[1-8]:LIMit:LOWer:LDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:USER1:TRACe8:LIMit:LOWer:LDATa 1,1601"
30 OUTPUT 717;":CALCulate:USER1:TRACe8:LIMit:LOWer:LDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

Description

This command clears lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.USER1.TRACe8.LIMit.LOWer.SEGMent.CLEar

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:LIMit:LOWer:SEGMent:CLEar

Example of use

10 OUTPUT 717;":CALCulate:USER1:TRACe8:LIMit:LOWer:SEGMent:CLEar"

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

Description

This command sets/gets number of lower limit segments, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Lower limit segments
Data Type	Long integer type (Long)
Range	1 ~ 100
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.USER1.TRACe8.LIMit.LOWer.SEGMent.COUNt = Var
Var = SCPI.CALCulate.USER1.TRACe8.LIMit.LOWer.SEGMent.COUNt
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:LIMit:LOWer:SEGMent:COUNt {1 ~ 100}

:CALCulate:USER[1-1]:TRACe[1-8]:LIMit:LOWer:SEGMent:COUNt?

Query Response

{1 ~ 100} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:LIMit:LOWer:SEGMent:COUNt 1 "
20 OUTPUT 717;":CALCulate:USER1:TRACe8:LIMit:LOWer:SEGMent:COUNt ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.DATA

Description

This command sets/gets lower limit line data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Lower limit data
Data Type	Variant type Array (Range)
Range	1...400
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 400) as Variant
SCPI.CALCulate.USER1.TRACe8.LIMit.LOWer.SEGMent.DATA = Var
Var = SCPI.CALCulate.USER1.TRACe8.LIMit.LOWer.SEGMent.DATA
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.LDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.CLEar

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.LOWer.SEGMent.COUNt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:LIMit:LOWer:SEGMent:DATA {1...400}

:CALCulate:USER[1-1]:TRACe[1-8]:LIMit:LOWer:SEGMent:DATA?

Query Response

{1...400} <newline><^END>

Example of use

```
10 Dim A(1:400)
20 OUTPUT 717;":CALCulate:USER1:TRACe8:LIMit:LOWer:SEGMent:DATA 1,400"
30 OUTPUT 717;":CALCulate:USER1:TRACe8:LIMit:LOWer:SEGMent:DATA ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.REPort.DATA

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.REPort.DATA

Description

This command gets pass/fail list, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test report
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
Var = SCPI.CALCulate.USER1.TRACe8.LIMit.REPort.DATA
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:LIMit:REPort[:DATA]?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:USER1:TRACe8:LIMit:REPort[:DATA] ?"
30 ENTER 717;A(*)
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.STATe = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.STATe

Description

This command sets/gets limit test on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Limit test (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.USER1.TRACe8.LIMit.STATe = Var
Var = SCPI.CALCulate.USER1.TRACe8.LIMit.STATe
```

Equivalent Key

USER Menu: **Display** > **Limit Test** > **Limit Test**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:LIMit[:STATe] {ON|OFF|1|0}

:CALCulate:USER[1-1]:TRACe[1-8]:LIMit[:STATe]?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:LIMit[:STATe] 1"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:LIMit[:STATe] ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.UPPer.LDATa

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.UPPer.LDATa = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.UPPer.LDATa

Description

This command sets/gets expanded upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Upper limit line
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.CALCulate.USER1.TRACe8.LIMit.UPPer.LDATa = Var
Var = SCPI.CALCulate.USER1.TRACe8.LIMit.UPPer.LDATa
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:LIMit:UPPer:LDATa {1...1601}

:CALCulate:USER[1-1]:TRACe[1-8]:LIMit:UPPer:LDATa?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":CALCulate:USER1:TRACe8:LIMit:UPPer:LDATa 1,1601"
30 OUTPUT 717;":CALCulate:USER1:TRACe8:LIMit:UPPer:LDATa ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

Description

This command clears upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.USER1.TRACe8.LIMit.UPPer.SEGMent.CLEar

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:LIMit:UPPer:SEGMent:CLEar

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:LIMit:UPPer:SEGMent:CLEar"
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Description

This command sets/gets number of upper limit segments, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Upper limit segment
Data Type	Long integer type (Long)
Range	1 ~ 100
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.CALCulate.USER1.TRACe8.LIMit.UPPer.SEGment.COUNt = Var
Var = SCPI.CALCulate.USER1.TRACe8.LIMit.UPPer.SEGment.COUNt
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.UPPer.SEGment.CLEar

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.UPPer.SEGment.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:LIMit:UPPer:SEGment:COUNt {1 ~ 100}

:CALCulate:USER[1-1]:TRACe[1-8]:LIMit:UPPer:SEGment:COUNt?

Query Response

{1 ~ 100} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:LIMit:UPPer:SEGment:COUNt 1 "
20 OUTPUT 717;":CALCulate:USER1:TRACe8:LIMit:UPPer:SEGment:COUNt ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.DATA

Description

This command sets/gets upper limit line data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Upper limit line data
Data Type	Variant type Array (Range)
Range	1...400
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var(1 to 400) as Variant
SCPI.CALCulate.USER1.TRACe8.LIMit.UPPer.SEGMent.DATA = Var
Var = SCPI.CALCulate.USER1.TRACe8.LIMit.UPPer.SEGMent.DATA
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.UPPer.LDATa

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.CLEar

SCPI.CALCulate.USER(Ch).TRACe(Tr).LIMit.UPPer.SEGMent.COUNt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:LIMit:UPPer:SEGMent:DATA {1...400}

:CALCulate:USER[1-1]:TRACe[1-8]:LIMit:UPPer:SEGMent:DATA?

Query Response

{1...400} <newline><^END>

Example of use

```
10 Dim A(1:400)
20 OUTPUT 717;":CALCulate:USER1:TRACe8:LIMit:UPPer:SEGMent:DATA 1,400"
30 OUTPUT 717;":CALCulate:USER1:TRACe8:LIMit:UPPer:SEGMent:DATA ?"
40 ENTER 717;A(*)
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

Description

This command sets marker search LPEak, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.USER1.TRACe8.MARKer10.SEARch.EXECute.LPEak

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

USER Menu: **Marker Search** > **Peak** > **Search Left**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:EXECute:LPEak

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:MARKer10:SEARch:EXECute:LPEak"
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

Description

This command sets marker search LTARget, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.USER1.TRACe8.MARKer10.SEARch.EXECute.LTARget

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

USER Menu: **Marker Search** > **Target** > **Search Left**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:EXECute:LTARget

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:MARKer10:SEARch:EXECute:LTARget"
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

Description

This command sets marker search MAXimum, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.USER1.TRACe8.MARKer10.SEARch.EXECute.MAXimum

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

USER Menu: **Marker Search** > **Search Max**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:EXECute:MAXimum

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:MARKer10:SEARch:EXECute:MAXimum"
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

Description

This command sets marker search MINimum, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.USER1.TRACe8.MARKer10.SEARch.EXECute.MINimum

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

USER Menu: **Marker Search** > **Search Min**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:EXECute:MINimum

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:MARKer10:SEARch:EXECute:MINimum"
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

Description

This command sets marker search PEAK, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.USER1.TRACe8.MARKer10.SEARch.EXECute.PEAK

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

USER Menu: **Marker Search** > **Peak** > **Search Peak**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:EXECute:PEAK

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:MARKer10:SEARch:EXECute:PEAK"
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

Description

This command sets marker search RPEak, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.USER1.TRACe8.MARKer10.SEARch.EXECute.RPEak

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

USER Menu: **Marker Search** > **Peak** > **Search Right**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:EXECute:RPEak

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:MARKer10:SEARch:EXECute:RPEak"
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

Description

This command sets marker search RTARget, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.USER1.TRACe8.MARKer10.SEARch.EXECute.RTARget

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Equivalent Key

USER Menu: **Marker Search** > **Target** > **Search Right**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:EXECute:RTARget

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:MARKer10:SEARch:EXECute:RTARget"
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.TARGet

Description

This command sets marker search TARGet, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Examples

SCPI.CALCulate.USER1.TRACe8.MARKer10.SEARch.EXECute.TARGet

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LPEak

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.LTARget

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MAXimum

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.MINimum

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.PEAK

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RPEak

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.EXECute.RTARget

Equivalent Key

USER Menu: **Marker Search** > **Target** > **Search Target**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:EXECute:TARGet

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:MARKer10:SEARch:EXECute:TARGet"
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion

Description

This command sets/gets marker-search-peak excursion value, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search peak excursion value
Data Type	Double precision floating point type (Double)
Range	0 ~ 10G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.CALCulate.USER1.TRACe8.MARKer10.SEARch.PEAK.EXCursion = Var
Var = SCPI.CALCulate.USER1.TRACe8.MARKer10.SEARch.PEAK.EXCursion
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity

Equivalent Key

USER Menu: **Marker Search** > **Peak** > **Peak Excursion**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:PEAK:EXCursion {0 ~ 10G}

:CALCulate:USER[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:PEAK:EXCursion?

Query Response

{0 ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:MARKer10:SEARch:PEAK:EXCursion 0"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:MARKer10:SEARch:PEAK:EXCursion ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.POLarity

Description

This command sets/gets marker-search-peak polarity type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search peak polarity type
Data Type	Character string type (String)
Range	POSitive|NEGative|BOTH
Preset Value	POSitive
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "POSitive"
SCPI.CALCulate.USER1.TRACe8.MARKer10.SEARch.PEAK.POLarity = Var
Var = SCPI.CALCulate.USER1.TRACe8.MARKer10.SEARch.PEAK.POLarity
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.PEAK.EXCursion

Equivalent Key

USER Menu: **Marker Search** > **Peak** > **Peak Polarity**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:PEAK:POLarity {POSitive|NEGative|BOTH}

:CALCulate:USER[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:PEAK:POLarity?

Query Response

{POSitive|NEGative|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:MARKer10:SEARch:PEAK:POLarity POSitive"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:MARKer10:SEARch:PEAK:POLarity ?"
30 ENTER 717;A$
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition

Description

This command sets/gets marker-target transition type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker target transition type
Data Type	Character string type (String)
Range	POSitive|NEGative|BOTH
Preset Value	BOTH
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as String
Var= "POSitive"
SCPI.CALCulate.USER1.TRACe8.MARKer10.SEARch.TARGet.TRANsition = Var
Var = SCPI.CALCulate.USER1.TRACe8.MARKer10.SEARch.TARGet.TRANsition
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y

Equivalent Key

USER Menu: **Marker Search** > **Target** > **Target Transition**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:TARGet:TRANsition {POSitive|NEGative|BOTH}

:CALCulate:USER[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:TARGet:TRANsition?

Query Response

{POSitive|NEGative|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:MARKer10:SEARch:TARGet:TRANsition POSitive"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:MARKer10:SEARch:TARGet:TRANsition ?"
30 ENTER 717;A$
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.Y

Description

This command sets/gets marker target value, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker target value
Data Type	Double precision floating point type (Double)
Range	-10G ~ 10G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -10000000000
SCPI.CALCulate.USER1.TRACe8.MARKer10.SEARch.TARGet.Y = Var
Var = SCPI.CALCulate.USER1.TRACe8.MARKer10.SEARch.TARGet.Y
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.TARGet.TRANsition

Equivalent Key

USER Menu: **Marker Search** > **Target** > **Target Value**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:TARGet:Y {-10G ~ 10G}

:CALCulate:USER[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:TARGet:Y?

Query Response

{-10G ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:MARKer10:SEARch:TARGet:Y -10000000000"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:MARKer10:SEARch:TARGet:Y ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).SEARch.TRACking.TYPE

Description

This command sets/gets search tracking type, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker search tracking type
Data Type	Character string type (String)
Range	OFF|MAXimum|MINimum|PEAK|TARGet
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.CALCulate.USER1.TRACe8.MARKer10.SEARch.TRACking.TYPE = Var
Var = SCPI.CALCulate.USER1.TRACe8.MARKer10.SEARch.TRACking.TYPE
```

Equivalent Key

USER Menu: **Marker Search** > **Tracking**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:TRACking:TYPE {OFF|MAXimum|MINimum|PEAK|TARGet}

:CALCulate:USER[1-1]:TRACe[1-8]:MARKer[1-10]:SEARch:TRACking:TYPE?

Query Response

{OFF|MAXimum|MINimum|PEAK|TARGet} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:MARKer10:SEARch:TRACking:TYPE OFF"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:MARKer10:SEARch:TRACking:TYPE ?"
30 ENTER 717;A$
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).STATe = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).STATe

Description

This command sets/gets marker visible on/off, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker visible state
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.USER1.TRACe8.MARKer10.STATe = Var
Var = SCPI.CALCulate.USER1.TRACe8.MARKer10.STATe
```

Equivalent Key

USER Menu: **Marker** > **Clear Marker Menu** > **Marker 1**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:MARKer[1-10]:STATe {ON|OFF|1|0}

:CALCulate:USER[1-1]:TRACe[1-8]:MARKer[1-10]:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:MARKer10:STATe 1"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:MARKer10:STATe ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).X

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).X = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).X

Description

This command sets/gets marker x position, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

NOTE Preset value may be clipped by SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STARt & SCPI.CALCulate.USER(Ch).TRACe(Tr).DATA.STOP

Variable

Parameter	*Value*
Description	Marker X position
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
SCPI.CALCulate.USER1.TRACe8.MARKer10.X = Var
Var = SCPI.CALCulate.USER1.TRACe8.MARKer10.X
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).Y

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:MARKer[1-10]:X

:CALCulate:USER[1-1]:TRACe[1-8]:MARKer[1-10]:X?

Query Response

{Double} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:MARKer10:X 0"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:MARKer10:X ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).Y

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).Y

Description

This command gets marker Y position, for the selected trace *Tr* and marker *Mk* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker Y position
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var = SCPI.CALCulate.USER1.TRACe8.MARKer10.Y
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).MARKer(Mk).X

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:MARKer[1-10]:Y?

Query Response

{Double} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:MARKer10:Y ?"
20 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).MATH.FUNCtion

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).MATH.FUNCtion = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).MATH.FUNCtion

Description

This command sets/gets math operation type, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Math function operation
Data Type	Character string type (String)
Range	NORMal|SUBTract|DIVide|ADD|MULTiply|USER
Preset Value	NORMal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NORMal"
SCPI.CALCulate.USER1.TRACe8.MATH.FUNCtion = Var
Var = SCPI.CALCulate.USER1.TRACe8.MATH.FUNCtion
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).MATH.MEMorize

SCPI.CALCulate.USER(Ch).TRACe(Tr).MATH.OFFSet

Equivalent Key

USER Menu: **Trace View** > **Data Math**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:MATH:FUNCtion {NORMal|SUBTract|DIVide|ADD|MULTiply|USER}

:CALCulate:USER[1-1]:TRACe[1-8]:MATH:FUNCtion?

Query Response

{NORMal|SUBTract|DIVide|ADD|MULTiply|USER} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:MATH:FUNCtion NORMal"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:MATH:FUNCtion ?"
30 ENTER 717;A$
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).MATH.MEMorize

Object Type

Method (**Write Only**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).MATH.MEMorize

Description

This command copies data to memory, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.CALCulate.USER1.TRACe8.MATH.MEMorize

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).MATH.FUNCtion

SCPI.CALCulate.USER(Ch).TRACe(Tr).MATH.OFFSet

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:MATH:MEMorize

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:MATH:MEMorize"
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).MATH.OFFSet

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).MATH.OFFSet = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).MATH.OFFSet

Description

This command sets/gets Trace Offset, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace offset
Data Type	Double precision floating point type (Double)
Range	-500G ~ 500G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -500000000000
SCPI.CALCulate.USER1.TRACe8.MATH.OFFSet = Var
Var = SCPI.CALCulate.USER1.TRACe8.MATH.OFFSet
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).MATH.FUNCtion

SCPI.CALCulate.USER(Ch).TRACe(Tr).MATH.MEMorize

Equivalent Key

USER Menu: **Trace View** > **Offset**

USER Menu: **Trace View** > **Marker** > **-Offset**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:MATH:OFFSet {-500G ~ 500G}

:CALCulate:USER[1-1]:TRACe[1-8]:MATH:OFFSet?

Query Response

{-500G ~ 500G} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:MATH:OFFSet -500000000000"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:MATH:OFFSet ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).SMOothing.APERture

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).SMOothing.APERture = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).SMOothing.APERture

Description

This command sets/gets Smoothing aperture, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Smoothing aperture
Data Type	Double precision floating point type (Double)
Range	50m ~ 25
Preset Value	1.5
Unit	%
Resolution	10m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0.05
SCPI.CALCulate.USER1.TRACe8.SMOothing.APERture = Var
Var = SCPI.CALCulate.USER1.TRACe8.SMOothing.APERture
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).SMOothing.STATe

Equivalent Key

USER Menu: **Trace View** > **Aperture**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:SMOothing:APERture {50m ~ 25}

:CALCulate:USER[1-1]:TRACe[1-8]:SMOothing:APERture?

Query Response

{50m ~ 25} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:SMOothing:APERture 0.05"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:SMOothing:APERture ?"
30 ENTER 717;A
```

SCPI.CALCulate.USER(Ch).TRACe(Tr).SMOothing.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.CALCulate.USER(Ch).TRACe(Tr).SMOothing.STATe = *Value*

Value = SCPI.CALCulate.USER(Ch).TRACe(Tr).SMOothing.STATe

Description

This command sets/gets Smoothing on/off, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Smoothing (On/OFF)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.CALCulate.USER1.TRACe8.SMOothing.STATe = Var
Var = SCPI.CALCulate.USER1.TRACe8.SMOothing.STATe
```

Related Objects

SCPI.CALCulate.USER(Ch).TRACe(Tr).SMOothing.APERture

Equivalent Key

USER Menu: **Trace View** > **Smoothing**

Equivalent SCPI Command

Syntax

:CALCulate:USER[1-1]:TRACe[1-8]:SMOothing:STATe {ON|OFF|1|0}

:CALCulate:USER[1-1]:TRACe[1-8]:SMOothing:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":CALCulate:USER1:TRACe8:SMOothing:STATe 1"
20 OUTPUT 717;":CALCulate:USER1:TRACe8:SMOothing:STATe ?"
30 ENTER 717;A
```

Control

SCPI.CONTrol.HANDler.A.DATA

Object Type

Method (**Write Only**)

Syntax

SCPI.CONTrol.HANDler.A.DATA = *Value*

Description

This command sets data to port A. 8-bit.

Variable

Parameter	*Value*
Description	Data to Port A
Data Type	Long integer type (Long)
Range	0 ~ 255
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.CONTrol.HANDler.A.DATA = Var
```

Related Objects

SCPI.CONTrol.HANDler.B.DATA

SCPI.CONTrol.HANDler.C.DATA

SCPI.CONTrol.HANDler.D.DATA

SCPI.CONTrol.HANDler.E.DATA

SCPI.CONTrol.HANDler.F.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CONTrol:HANDler:A[:DATA] {0 ~ 255}

Query Response

{0 ~ 255} <newline><^END>

Example of use

```
10 OUTPUT 717;":CONTrol:HANDler:A[:DATA] 0 "
20 ENTER 717;A
```

SCPI.CONTrol.HANDler.B.DATA

Object Type

Method (**Write Only**)

Syntax

SCPI.CONTrol.HANDler.B.DATA = *Value*

Description

This command sets data to port B. 8-bit.

Variable

Parameter	*Value*
Description	Data to Port B
Data Type	Long integer type (Long)
Range	0 ~ 255
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.CONTrol.HANDler.B.DATA = Var
```

Related Objects

SCPI.CONTrol.HANDler.A.DATA

SCPI.CONTrol.HANDler.C.DATA

SCPI.CONTrol.HANDler.D.DATA

SCPI.CONTrol.HANDler.E.DATA

SCPI.CONTrol.HANDler.F.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CONTrol:HANDler:B[:DATA] {0 ~ 255}

Query Response

{0 ~ 255} <newline><^END>

Example of use

```
10 OUTPUT 717;":CONTrol:HANDler:B[:DATA] 0 "
20 ENTER 717;A
```

SCPI.CONTrol.HANDler.C.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CONTrol.HANDler.C.DATA = *Value*

Value = SCPI.CONTrol.HANDler.C.DATA

Description

This command sets/gets data to/from port C.

Variable

Parameter	*Value*
Description	Data to Port C
Data Type	Long integer type (Long)
Range	0 ~ 15
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.CONTrol.HANDler.C.DATA = Var
Var = SCPI.CONTrol.HANDler.C.DATA
```

Related Objects

SCPI.CONTrol.HANDler.A.DATA

SCPI.CONTrol.HANDler.B.DATA

SCPI.CONTrol.HANDler.D.DATA

SCPI.CONTrol.HANDler.E.DATA

SCPI.CONTrol.HANDler.F.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CONTrol:HANDler:C[:DATA] {0 ~ 15}

:CONTrol:HANDler:C[:DATA]?

Query Response

{0 ~ 15} <newline><^END>

Example of use

```
10 OUTPUT 717;":CONTrol:HANDler:C[:DATA] 0 "
20 OUTPUT 717;":CONTrol:HANDler:C[:DATA] ?"
30 ENTER 717;A
```

SCPI.CONTrol.HANDler.C.MODE

Object Type

Property (**Read-Write**)

Syntax

SCPI.CONTrol.HANDler.C.MODE = *Value*

Value = SCPI.CONTrol.HANDler.C.MODE

Description

This command sets/gets direction of port C. 4-bit.

Variable

Parameter	*Value*
Description	Direction of Port C
Data Type	Character string type (String)
Range	INPut|OUTPut
Preset Value	INPut
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as String
Var= "INPut"
SCPI.CONTrol.HANDler.C.MODE = Var
Var = SCPI.CONTrol.HANDler.C.MODE
```

Related Objects

SCPI.CONTrol.HANDler.D.MODE

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CONTrol:HANDler:C:MODE {INPut|OUTPut}

:CONTrol:HANDler:C:MODE?

Query Response

{INPut|OUTPut} <newline><^END>

Example of use

```
10 OUTPUT 717;":CONTrol:HANDler:C:MODE INPut"
20 OUTPUT 717;":CONTrol:HANDler:C:MODE ?"
30 ENTER 717;A$
```

SCPI.CONTrol.HANDler.D.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CONTrol.HANDler.D.DATA = *Value*

Value = SCPI.CONTrol.HANDler.D.DATA

Description

This command sets/gets data to/from port D. 4-bit.

Variable

Parameter	*Value*
Description	Data to Port D
Data Type	Long integer type (Long)
Range	0 ~ 15
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.CONTrol.HANDler.D.DATA = Var
Var = SCPI.CONTrol.HANDler.D.DATA
```

Related Objects

SCPI.CONTrol.HANDler.A.DATA

SCPI.CONTrol.HANDler.B.DATA

SCPI.CONTrol.HANDler.C.DATA

SCPI.CONTrol.HANDler.E.DATA

SCPI.CONTrol.HANDler.F.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CONTrol:HANDler:D[:DATA] {0 ~ 15}

:CONTrol:HANDler:D[:DATA]?

Query Response

{0 ~ 15} <newline><^END>

Example of use

```
10 OUTPUT 717;":CONTrol:HANDler:D[:DATA] 0 "
20 OUTPUT 717;":CONTrol:HANDler:D[:DATA] ?"
30 ENTER 717;A
```

SCPI.CONTrol.HANDler.D.MODE

Object Type

Property (**Read-Write**)

Syntax

SCPI.CONTrol.HANDler.D.MODE = *Value*

Value = SCPI.CONTrol.HANDler.D.MODE

Description

This command sets/gets direction of port D.

Variable

Parameter	*Value*
Description	Direction of Port D
Data Type	Character string type (String)
Range	INPut|OUTPut
Preset Value	INPut
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as String
Var= "INPut"
SCPI.CONTrol.HANDler.D.MODE = Var
Var = SCPI.CONTrol.HANDler.D.MODE
```

Related Objects

SCPI.CONTrol.HANDler.C.MODE

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CONTrol:HANDler:D:MODE {INPut|OUTPut}

:CONTrol:HANDler:D:MODE?

Query Response

{INPut|OUTPut} <newline><^END>

Example of use

```
10 OUTPUT 717;":CONTrol:HANDler:D:MODE INPut"
20 OUTPUT 717;":CONTrol:HANDler:D:MODE ?"
30 ENTER 717;A$
```

SCPI.CONTrol.HANDler.E.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CONTrol.HANDler.E.DATA = *Value*

Value = SCPI.CONTrol.HANDler.E.DATA

Description

This command sets/gets data to/from port E. (C Port + D Port) 8-bit.

Variable

Parameter	*Value*
Description	Data to Port E
Data Type	Long integer type (Long)
Range	0 ~ 255
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.CONTrol.HANDler.E.DATA = Var
Var = SCPI.CONTrol.HANDler.E.DATA
```

Related Objects

SCPI.CONTrol.HANDler.A.DATA

SCPI.CONTrol.HANDler.B.DATA

SCPI.CONTrol.HANDler.C.DATA

SCPI.CONTrol.HANDler.D.DATA

SCPI.CONTrol.HANDler.F.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CONTrol:HANDler:E[:DATA] {0 ~ 255}

:CONTrol:HANDler:E[:DATA]?

Query Response

{0 ~ 255} <newline><^END>

Example of use

```
10 OUTPUT 717;":CONTrol:HANDler:E[:DATA] 0 "
20 OUTPUT 717;":CONTrol:HANDler:E[:DATA] ?"
30 ENTER 717;A
```

SCPI.CONTrol.HANDler.F.DATA

Object Type

Method (**Write Only**)

Syntax

SCPI.CONTrol.HANDler.F.DATA = *Value*

Description

This command sets data to port F. (A Port + B Port) 16-bit.

Variable

Parameter	*Value*
Description	Data to port F
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.CONTrol.HANDler.F.DATA = Var
```

Related Objects

SCPI.CONTrol.HANDler.A.DATA

SCPI.CONTrol.HANDler.B.DATA

SCPI.CONTrol.HANDler.C.DATA

SCPI.CONTrol.HANDler.D.DATA

SCPI.CONTrol.HANDler.E.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CONTrol:HANDler:F[:DATA] {0 ~ 65535}

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":CONTrol:HANDler:F[:DATA] 0 "
20 ENTER 717;A
```

SCPI.CONTrol.HANDler.OUTPut(1-2).DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.CONTrol.HANDler.OUTPut(1-2).DATA = *Value*

Value = SCPI.CONTrol.HANDler.OUTPut(1-2).DATA

Description

This command sets/gets data to OUTPUT1 or OUTPUT2..

Variable

Parameter	*Value*
Description	Data to Output1 & Output2
Data Type	Long integer type (Long)
Range	0 ~ 1
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.CONTrol.HANDler.OUTPut2.DATA = Var
Var = SCPI.CONTrol.HANDler.OUTPut2.DATA
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:CONTrol:HANDler:OUTPut[1-2][:DATA] {0 ~ 1}

:CONTrol:HANDler:OUTPut[1-2][:DATA]?

Query Response

{0 ~ 1} <newline><^END>

Example of use

```
10 OUTPUT 717;":CONTrol:HANDler:OUTPut2[:DATA] 0 "
20 OUTPUT 717;":CONTrol:HANDler:OUTPut2[:DATA] ?"
30 ENTER 717;A
```

Display

SCPI.DISPlay.AM(Ch).ALLTrace.PERSistence.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.AM(Ch).ALLTrace.PERSistence.CLEar

Description

This command clears all stored traces, for the selected channel *Ch*.

Examples

SCPI.DISPlay.AM1.ALLTrace.PERSistence.CLEar

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:ALLTrace:PERSistence:CLEar

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:ALLTrace:PERSistence:CLEar"
```

SCPI.DISPlay.AM(Ch).ANNotation.MARKer.POSition

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.AM(Ch).ANNotation.MARKer.POSition = *Value*

Value = SCPI.DISPlay.AM(Ch).ANNotation.MARKer.POSition

Description

This command sets/gets Marker annotation position, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker annotation position
Data Type	Character string type (String)
Range	LEFT|RIGHt
Preset Value	RIGHt
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "LEFT"
SCPI.DISPlay.AM1.ANNotation.MARKer.POSition = Var
Var = SCPI.DISPlay.AM1.ANNotation.MARKer.POSition
```

Related Objects

SCPI.DISPlay.AM(Ch).ANNotation.MEASurement.STATe

Equivalent Key

AM Menu: **Display** > **Marker Information**

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:ANNotation:MARKer:POSition {LEFT|RIGHt}

:DISPlay:AM[1-1]:ANNotation:MARKer:POSition?

Query Response

{LEFT|RIGHt} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:ANNotation:MARKer:POSition LEFT"
20 OUTPUT 717;":DISPlay:AM1:ANNotation:MARKer:POSition ?"
30 ENTER 717;A$
```

SCPI.DISPlay.AM(Ch).ANNotation.MEASurement.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.AM(Ch).ANNotation.MEASurement.STATe = *Value*

Value = SCPI.DISPlay.AM(Ch).ANNotation.MEASurement.STATe

Description

This command sets/gets Show/Hide measurement setting, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/Hide measurement setting
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.AM1.ANNotation.MEASurement.STATe = Var
Var = SCPI.DISPlay.AM1.ANNotation.MEASurement.STATe
```

Related Objects

SCPI.DISPlay.AM(Ch).ANNotation.MARKer.POSition

Equivalent Key

AM Menu: **Display** > **Meas Condition**

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:ANNotation:MEASurement:STATe {ON|OFF|1|0}

:DISPlay:AM[1-1]:ANNotation:MEASurement:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:ANNotation:MEASurement:STATe 1"
20 OUTPUT 717;":DISPlay:AM1:ANNotation:MEASurement:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.AM(Ch).GRATicule.AXIS.Y.RELative

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.AM(Ch).GRATicule.AXIS.Y.RELative = *Value*

Value = SCPI.DISPlay.AM(Ch).GRATicule.AXIS.Y.RELative

Description

This command sets/gets force graticule label notation relative. If OFF, absolute notation is used if possible, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Force graticule label notation
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.AM1.GRATicule.AXIS.Y.RELative = Var
Var = SCPI.DISPlay.AM1.GRATicule.AXIS.Y.RELative
```

Related Objects

SCPI.DISPlay.AM(Ch).GRATicule.AXIS.Y.STATe

Equivalent Key

AM Menu: **Display** > **Relative Y-Scale**

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:GRATicule:AXIS:Y:RELative {ON|OFF|1|0}

:DISPlay:AM[1-1]:GRATicule:AXIS:Y:RELative?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:GRATicule:AXIS:Y:RELative 1"
20 OUTPUT 717;":DISPlay:AM1:GRATicule:AXIS:Y:RELative ?"
30 ENTER 717;A
```

SCPI.DISPlay.AM(Ch).GRATicule.AXIS.Y.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.AM(Ch).GRATicule.AXIS.Y.STATe = *Value*

Value = SCPI.DISPlay.AM(Ch).GRATicule.AXIS.Y.STATe

Description

This command sets/gets Show/Hide Y graticule label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/Hide Y graticule label
Data Type	Character string type (String)
Range	OFF|SHORt|MIDDle|LONG
Preset Value	SHORt
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.DISPlay.AM1.GRATicule.AXIS.Y.STATe = Var
Var = SCPI.DISPlay.AM1.GRATicule.AXIS.Y.STATe
```

Related Objects

SCPI.DISPlay.AM(Ch).GRATicule.AXIS.Y.RELative

Equivalent Key

AM Menu: **Display** > **Y # of Digits**

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:GRATicule:AXIS:Y:STATe {OFF|SHORt|MIDDle|LONG}

:DISPlay:AM[1-1]:GRATicule:AXIS:Y:STATe?

Query Response

{OFF|SHORt|MIDDle|LONG} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:GRATicule:AXIS:Y:STATe OFF"
20 OUTPUT 717;":DISPlay:AM1:GRATicule:AXIS:Y:STATe ?"
30 ENTER 717;A$
```

SCPI.DISPlay.AM(Ch).LABel.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.AM(Ch).LABel.DATA = *Value*

Value = SCPI.DISPlay.AM(Ch).LABel.DATA

Description

This command sets/gets Window Title Label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Window Title Label
Data Type	Character string type (String)
Range	254 chars
Preset Value	""
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
SCPI.DISPlay.AM1.LABel.DATA = Var
Var = SCPI.DISPlay.AM1.LABel.DATA
```

Related Objects

SCPI.DISPlay.AM(Ch).LABel.STATe

Equivalent Key

AM Menu: **Display** > **Edit Title Label**

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:LABel:DATA

:DISPlay:AM[1-1]:LABel:DATA?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:LABel:DATA Sample_Text"
20 OUTPUT 717;":DISPlay:AM1:LABel:DATA ?"
30 ENTER 717;A$
```

SCPI.DISPlay.AM(Ch).LABel.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.AM(Ch).LABel.STATe = *Value*

Value = SCPI.DISPlay.AM(Ch).LABel.STATe

Description

This command sets/gets Show/Hide Window Title Label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/Hide Window Title Label
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.AM1.LABel.STATe = Var
Var = SCPI.DISPlay.AM1.LABel.STATe
```

Related Objects

SCPI.DISPlay.AM(Ch).LABel.DATA

Equivalent Key

AM Menu: **Display** > **Title Label**

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:LABel:STATe {ON|OFF|1|0}

:DISPlay:AM[1-1]:LABel:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:LABel:STATe 1"
20 OUTPUT 717;":DISPlay:AM1:LABel:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.AM(Ch).LIMit.FSIGn

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.AM(Ch).LIMit.FSIGn = *Value*

Value = SCPI.DISPlay.AM(Ch).LIMit.FSIGn

Description

This command sets/gets show/hide fail sign, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/hide fail sign
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.AM1.LIMit.FSIGn = Var
Var = SCPI.DISPlay.AM1.LIMit.FSIGn
```

Equivalent Key

AM Menu: **Display** > **Limit Test** > **Fail Sign**

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:LIMit:FSIGn {ON|OFF|1|0}

:DISPlay:AM[1-1]:LIMit:FSIGn?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:LIMit:FSIGn 1"
20 OUTPUT 717;":DISPlay:AM1:LIMit:FSIGn ?"
30 ENTER 717;A
```

SCPI.DISPlay.AM(Ch).MAXimize

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.AM(Ch).MAXimize = *Value*

Value = SCPI.DISPlay.AM(Ch).MAXimize

Description

This command sets/gets maximize active trace, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Maximize active trace
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.AM1.MAXimize = Var
Var = SCPI.DISPlay.AM1.MAXimize
```

Related Objects

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:MAXimize {ON|OFF|1|0}

:DISPlay:AM[1-1]:MAXimize?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:MAXimize 1"
20 OUTPUT 717;":DISPlay:AM1:MAXimize ?"
30 ENTER 717;A
```

SCPI.DISPlay.AM(Ch).STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.AM(Ch).STATe = *Value*

Value = SCPI.DISPlay.AM(Ch).STATe

Description

This command sets/gets measurement display on/off. At least one measurement window must be turned on, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	measurement display on/off
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	60, "Failed to hide window" 100, "Option not installed"

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.AM1.STATe = Var
Var = SCPI.DISPlay.AM1.STATe
```

Equivalent Key

PN Menu: **Measurement View** > **Show Window** > **AM Noise**

SP Menu: **Measurement View** > **Show Window** > **AM Noise**

FP Menu: **Measurement View** > **Show Window** > **AM Noise**

TR Menu: **Measurement View** > **Show Window** > **AM Noise**

AM Menu: **Measurement View** > **Show Window** > **AM Noise**

BB Menu: **Measurement View** > **Show Window** > **AM Noise**

USER Menu: **Measurement View** > **Show Window** > **AM Noise**

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:STATe {ON|OFF|1|0}

:DISPlay:AM[1-1]:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:STATe 1"
20 OUTPUT 717;":DISPlay:AM1:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.AM(Ch).TABLe.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.AM(Ch).TABLe.STATe = *Value*

Value = SCPI.DISPlay.AM(Ch).TABLe.STATe

Description

This command sets/gets show/hide text area, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/hide text area
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.AM1.TABLe.STATe = Var
Var = SCPI.DISPlay.AM1.TABLe.STATe
```

Related Objects

Equivalent Key

AM Menu: **Marker** > **Marker List**

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:TABLe[:STATe] {ON|OFF|1|0}

:DISPlay:AM[1-1]:TABLe[:STATe]?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:TABLe[:STATe] 1"
20 OUTPUT 717;":DISPlay:AM1:TABLe[:STATe] ?"
30 ENTER 717;A
```

SCPI.DISPlay.AM(Ch).TRACe(Tr).LABel.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.AM(Ch).TRACe(Tr).LABel.DATA = *Value*

Value = SCPI.DISPlay.AM(Ch).TRACe(Tr).LABel.DATA

Description

This command sets/gets Trace Title Label, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace Title Label
Data Type	Character string type (String)
Range	254 chars
Preset Value	"AM Noise"
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
SCPI.DISPlay.AM1.TRACe1.LABel.DATA = Var
Var = SCPI.DISPlay.AM1.TRACe1.LABel.DATA
```

Equivalent Key

AM Menu: **Trace View** > **Trace Label**

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:TRACe[1-1]:LABel:DATA

:DISPlay:AM[1-1]:TRACe[1-1]:LABel:DATA?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:TRACe1:LABel:DATA Sample_Text"
20 OUTPUT 717;":DISPlay:AM1:TRACe1:LABel:DATA ?"
30 ENTER 717;A$
```

SCPI.DISPlay.AM(Ch).TRACe(Tr).LIMit.LINE

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.AM(Ch).TRACe(Tr).LIMit.LINE = *Value*

Value = SCPI.DISPlay.AM(Ch).TRACe(Tr).LIMit.LINE

Description

This command sets/gets show/hide limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/hide limit line
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.AM1.TRACe1.LIMit.LINE = Var
Var = SCPI.DISPlay.AM1.TRACe1.LIMit.LINE
```

Equivalent Key

AM Menu: **Display** > **Limit Test** > **Limit Line**

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:TRACe[1-1]:LIMit:LINE {ON|OFF|1|0}

:DISPlay:AM[1-1]:TRACe[1-1]:LIMit:LINE?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:TRACe1:LIMit:LINE 1"
20 OUTPUT 717;":DISPlay:AM1:TRACe1:LIMit:LINE ?"
30 ENTER 717;A
```

SCPI.DISPlay.AM(Ch).TRACe(Tr).MODE

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.AM(Ch).TRACe(Tr).MODE = *Value*

Value = SCPI.DISPlay.AM(Ch).TRACe(Tr).MODE

Description

This command sets/gets show data and/or memory trace, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show data and/or memory trace
Data Type	Character string type (String)
Range	OFF|DATA|MEMory|BOTH
Preset Value	DATA
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.DISPlay.AM1.TRACe1.MODE = Var
Var = SCPI.DISPlay.AM1.TRACe1.MODE
```

Equivalent Key

AM Menu: **Trace View** > **Display Trace**

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:TRACe[1-1]:MODE {OFF|DATA|MEMory|BOTH}

:DISPlay:AM[1-1]:TRACe[1-1]:MODE?

Query Response

{OFF|DATA|MEMory|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:TRACe1:MODE OFF"
20 OUTPUT 717;":DISPlay:AM1:TRACe1:MODE ?"
30 ENTER 717;A$
```

SCPI.DISPlay.AM(Ch).TRACe(Tr).PERSistence.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.AM(Ch).TRACe(Tr).PERSistence.CLEar

Description

This command clears stored traces, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.DISPlay.AM1.TRACe1.PERSistence.CLEar

Related Objects

SCPI.DISPlay.AM(Ch).TRACe(Tr).PERSistence.STATe

Equivalent Key

AM Menu: **Trace View** > **Persistence** > **Clear Persistent Data**

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:TRACe[1-1]:PERSistence:CLEar

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:TRACe1:PERSistence:CLEar"
```

SCPI.DISPlay.AM(Ch).TRACe(Tr).PERSistence.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.AM(Ch).TRACe(Tr).PERSistence.STATe = *Value*

Value = SCPI.DISPlay.AM(Ch).TRACe(Tr).PERSistence.STATe

Description

This command sets/gets trace storage mode, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace storage mode
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.AM1.TRACe1.PERSistence.STATe = Var
Var = SCPI.DISPlay.AM1.TRACe1.PERSistence.STATe
```

Related Objects

SCPI.DISPlay.AM(Ch).TRACe(Tr).PERSistence.CLEar

Equivalent Key

AM Menu: **Trace View** > **Persistence** > **Persistence Mode**

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:TRACe[1-1]:PERSistence:STATe {ON|OFF|1|0}

:DISPlay:AM[1-1]:TRACe[1-1]:PERSistence:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:TRACe1:PERSistence:STATe 1"
20 OUTPUT 717;":DISPlay:AM1:TRACe1:PERSistence:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.AM(Ch).TRACe(Tr).X.SCALe.AUTO

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.AM(Ch).TRACe(Tr).X.SCALe.AUTO = *Value*

Value = SCPI.DISPlay.AM(Ch).TRACe(Tr).X.SCALe.AUTO

Description

This command sets/gets autoscale setting for X axis, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Autoscale setting
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.AM1.TRACe1.X.SCALe.AUTO = Var
Var = SCPI.DISPlay.AM1.TRACe1.X.SCALe.AUTO
```

Related Objects

SCPI.DISPlay.AM(Ch).TRACe(Tr).X.SCALe.AUTO

SCPI.DISPlay.AM(Ch).TRACe(Tr).X.SCALe.LEFT

SCPI.DISPlay.AM(Ch).TRACe(Tr).X.SCALe.RIGHt

Equivalent Key

AM Menu: **Scale** > **X Axis** > **Auto**

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:TRACe[1-1]:X[:SCALe]:AUTO {ON|OFF|1|0}

:DISPlay:AM[1-1]:TRACe[1-1]:X[:SCALe]:AUTO?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:TRACe1:X[:SCALe]:AUTO 1"
20 OUTPUT 717;":DISPlay:AM1:TRACe1:X[:SCALe]:AUTO ?"
30 ENTER 717;A
```

SCPI.DISPlay.AM(Ch).TRACe(Tr).X.SCALe.LEFT

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.AM(Ch).TRACe(Tr).X.SCALe.LEFT = *Value*

Value = SCPI.DISPlay.AM(Ch).TRACe(Tr).X.SCALe.LEFT

Description

This command sets/gets left setting for X axis, for the selected trace *Tr* of the selected channel *Ch*.

NOTE This command is available only when the automatic setting of the X-axis display range (set with SCPI.DISPlay.AM(Ch).TRACe(Tr).X.SCALe.AUTO) is set to OFF.

Variable

Parameter	*Value*
Description	left setting for X axis
Data Type	Double precision floating point type (Double)
Range	1 ~ 39.9999999M
Preset Value	1k
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.DISPlay.AM1.TRACe1.X.SCALe.LEFT = Var
Var = SCPI.DISPlay.AM1.TRACe1.X.SCALe.LEFT
```

Related Objects

SCPI.DISPlay.AM(Ch).TRACe(Tr).X.SCALe.AUTO

SCPI.DISPlay.AM(Ch).TRACe(Tr).X.SCALe.RIGHt

Equivalent Key

AM Menu: **Scale** > **X Axis** > **Left**

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:TRACe[1-1]:X[:SCALe]:LEFT {1 ~ 39.9999999M}

:DISPlay:AM[1-1]:TRACe[1-1]:X[:SCALe]:LEFT?

Query Response

{1 ~ 39.9999999M} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:TRACe1:X[:SCALe]:LEFT 0"
20 OUTPUT 717;":DISPlay:AM1:TRACe1:X[:SCALe]:LEFT ?"
30 ENTER 717;A
```

SCPI.DISPlay.AM(Ch).TRACe(Tr).X.SCALe.RIGHt

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.AM(Ch).TRACe(Tr).X.SCALe.RIGHt = *Value*

Value = SCPI.DISPlay.AM(Ch).TRACe(Tr).X.SCALe.RIGHt

Description

This command sets/gets right setting for X axis, for the selected trace *Tr* of the selected channel *Ch*.

NOTE This command is available only when the automatic setting of the X-axis display range (set with SCPI.DISPlay.AM(Ch).TRACe(Tr).X.SCALe.AUTO) is set to OFF.

Variable

Parameter	*Value*
Description	Right setting for X axis
Data Type	Double precision floating point type (Double)
Range	1.1 ~ 40M
Preset Value	10M
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.DISPlay.AM1.TRACe1.X.SCALe.RIGHt = Var
Var = SCPI.DISPlay.AM1.TRACe1.X.SCALe.RIGHt
```

Related Objects

SCPI.DISPlay.AM(Ch).TRACe(Tr).X.SCALe.AUTO

SCPI.DISPlay.AM(Ch).TRACe(Tr).X.SCALe.LEFT

Equivalent Key

AM Menu: **Scale** > **X Axis** > **Right**

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:TRACe[1-1]:X[:SCALe]:RIGHt {1.1 ~ 40M}

:DISPlay:AM[1-1]:TRACe[1-1]:X[:SCALe]:RIGHt?

Query Response

{1.1 ~ 40M} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:TRACe1:X[:SCALe]:RIGHt 0"
20 OUTPUT 717;":DISPlay:AM1:TRACe1:X[:SCALe]:RIGHt ?"
30 ENTER 717;A
```

SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.AUTO

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.AUTO

Description

This command sets auto scale setting for Y axis, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.DISPlay.AM1.TRACe1.Y.SCALe.AUTO

Related Objects

SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.RLEVel

SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.RPOSition

Equivalent Key

AM Menu: **Scale** > **Auto Scale**

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:TRACe[1-1]:Y[:SCALe]:AUTO

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:TRACe1:Y[:SCALe]:AUTO"
```

SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.PDIVision

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.PDIVision = *Value*

Value = SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.PDIVision

Description

This command sets/gets scale per division, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale per division
Data Type	Double precision floating point type (Double)
Range	1a ~ 10G
Preset Value	10
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0.000000000001
SCPI.DISPlay.AM1.TRACe1.Y.SCALe.PDIVision = Var
Var = SCPI.DISPlay.AM1.TRACe1.Y.SCALe.PDIVision
```

Related Objects

SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.RLEVel

SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.RPOSition

Equivalent Key

AM Menu: **Scale** > **Scale/Div**

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:TRACe[1-1]:Y[:SCALe]:PDIVision {1a ~ 10G}

:DISPlay:AM[1-1]:TRACe[1-1]:Y[:SCALe]:PDIVision?

Query Response

{1a ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:TRACe1:Y[:SCALe]:PDIVision 0.000000000001"
20 OUTPUT 717;":DISPlay:AM1:TRACe1:Y[:SCALe]:PDIVision ?"
30 ENTER 717;A
```

SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.RLEVel

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.RLEVel = *Value*

Value = SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.RLEVel

Description

This command sets/gets scale reference level, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale reference level
Data Type	Double precision floating point type (Double)
Range	-50G ~ 50G
Preset Value	-20
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -50000000000
SCPI.DISPlay.AM1.TRACe1.Y.SCALe.RLEVel = Var
Var = SCPI.DISPlay.AM1.TRACe1.Y.SCALe.RLEVel
```

Related Objects

SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.RPOSition

Equivalent Key

AM Menu: **Scale** > **Reference Value**

AM Menu: **Scale** > **Marker** > **Reference**

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:TRACe[1-1]:Y[:SCALe]:RLEVel {-50G ~ 50G}

:DISPlay:AM[1-1]:TRACe[1-1]:Y[:SCALe]:RLEVel?

Query Response

{-50G ~ 50G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:TRACe1:Y[:SCALe]:RLEVel -50000000000"
20 OUTPUT 717;":DISPlay:AM1:TRACe1:Y[:SCALe]:RLEVel ?"
30 ENTER 717;A
```

SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.RPOSition

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.RPOSition = *Value*

Value = SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.RPOSition

Description

This command sets/gets scale reference position, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale reference position
Data Type	Long integer type (Long)
Range	0 ~ 30
Preset Value	12
Unit	Div
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.DISPlay.AM1.TRACe1.Y.SCALe.RPOSition = Var
Var = SCPI.DISPlay.AM1.TRACe1.Y.SCALe.RPOSition
```

Related Objects

SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.AM(Ch).TRACe(Tr).Y.SCALe.RLEVel

Equivalent Key

AM Menu: **Scale** > **Reference Position**

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:TRACe[1-1]:Y[:SCALe]:RPOSition {0 ~ 30}

:DISPlay:AM[1-1]:TRACe[1-1]:Y[:SCALe]:RPOSition?

Query Response

{0 ~ 30} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:TRACe1:Y[:SCALe]:RPOSition 0 "
20 OUTPUT 717;":DISPlay:AM1:TRACe1:Y[:SCALe]:RPOSition ?"
30 ENTER 717;A
```

SCPI.DISPlay.AM(Ch).Y.SCALe.DIVisions

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.AM(Ch).Y.SCALe.DIVisions = *Value*

Value = SCPI.DISPlay.AM(Ch).Y.SCALe.DIVisions

Description

This command sets/gets number of Y division, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Number of Y division
Data Type	Long integer type (Long)
Range	4 ~ 30
Preset Value	12
Unit	-
Resolution	2
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 4
SCPI.DISPlay.AM1.Y.SCALe.DIVisions = Var
Var = SCPI.DISPlay.AM1.Y.SCALe.DIVisions
```

Equivalent Key

AM Menu: **Scale** > **Divisions**

Equivalent SCPI Command

Syntax

:DISPlay:AM[1-1]:Y[:SCALe]:DIVisions {4 ~ 30}

:DISPlay:AM[1-1]:Y[:SCALe]:DIVisions?

Query Response

{4 ~ 30} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:AM1:Y[:SCALe]:DIVisions 4 "
20 OUTPUT 717;":DISPlay:AM1:Y[:SCALe]:DIVisions ?"
30 ENTER 717;A
```

SCPI.DISPlay.BB(Ch).ALLTrace.PERSistence.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.BB(Ch).ALLTrace.PERSistence.CLEar

Description

This command clears all stored traces, for the selected channel *Ch*.

Examples

SCPI.DISPlay.BB1.ALLTrace.PERSistence.CLEar

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:ALLTrace:PERSistence:CLEar

Example of use

10 OUTPUT 717;":DISPlay:BB1:ALLTrace:PERSistence:CLEar"

SCPI.DISPlay.BB(Ch).ANNotation.MARKer.POSition

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.BB(Ch).ANNotation.MARKer.POSition = *Value*

Value = SCPI.DISPlay.BB(Ch).ANNotation.MARKer.POSition

Description

This command sets/gets Marker annotation position, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker annotation position
Data Type	Character string type (String)
Range	LEFT|RIGHt
Preset Value	RIGHt
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "LEFT"
SCPI.DISPlay.BB1.ANNotation.MARKer.POSition = Var
Var = SCPI.DISPlay.BB1.ANNotation.MARKer.POSition
```

Related Objects

SCPI.DISPlay.BB(Ch).ANNotation.MEASurement.STATe

Equivalent Key

BB Menu: **Display** > **Marker Information**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:ANNotation:MARKer:POSition {LEFT|RIGHt}

:DISPlay:BB[1-1]:ANNotation:MARKer:POSition?

Query Response

{LEFT|RIGHt} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:ANNotation:MARKer:POSition LEFT"
20 OUTPUT 717;":DISPlay:BB1:ANNotation:MARKer:POSition ?"
30 ENTER 717;A$
```

SCPI.DISPlay.BB(Ch).ANNotation.MEASurement.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.BB(Ch).ANNotation.MEASurement.STATe = *Value*

Value = SCPI.DISPlay.BB(Ch).ANNotation.MEASurement.STATe

Description

This command sets/gets Show/Hide measurement setting, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/Hide measurement setting
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.BB1.ANNotation.MEASurement.STATe = Var
Var = SCPI.DISPlay.BB1.ANNotation.MEASurement.STATe
```

Related Objects

SCPI.DISPlay.BB(Ch).ANNotation.MARKer.POSition

Equivalent Key

BB Menu: **Display** > **Meas Condition**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:ANNotation:MEASurement:STATe {ON|OFF|1|0}

:DISPlay:BB[1-1]:ANNotation:MEASurement:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:ANNotation:MEASurement:STATe 1"
20 OUTPUT 717;":DISPlay:BB1:ANNotation:MEASurement:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.BB(Ch).GRATicule.AXIS.Y.RELative

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.BB(Ch).GRATicule.AXIS.Y.RELative = *Value*

Value = SCPI.DISPlay.BB(Ch).GRATicule.AXIS.Y.RELative

Description

This command sets/gets force graticule label notation relative. If OFF, absolute notation is used if possible, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Force graticule label notation
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.BB1.GRATicule.AXIS.Y.RELative = Var
Var = SCPI.DISPlay.BB1.GRATicule.AXIS.Y.RELative
```

Related Objects

SCPI.DISPlay.BB(Ch).GRATicule.AXIS.Y.STATe

Equivalent Key

BB Menu: **Display** > **Relative Y-Scale**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:GRATicule:AXIS:Y:RELative {ON|OFF|1|0}

:DISPlay:BB[1-1]:GRATicule:AXIS:Y:RELative?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:GRATicule:AXIS:Y:RELative 1"
20 OUTPUT 717;":DISPlay:BB1:GRATicule:AXIS:Y:RELative ?"
30 ENTER 717;A
```

SCPI.DISPlay.BB(Ch).GRATicule.AXIS.Y.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.BB(Ch).GRATicule.AXIS.Y.STATe = *Value*

Value = SCPI.DISPlay.BB(Ch).GRATicule.AXIS.Y.STATe

Description

This command sets/gets Show/Hide Y graticule label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/Hide Y graticule label
Data Type	Character string type (String)
Range	OFF|SHORt|MIDDle|LONG
Preset Value	SHORt
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.DISPlay.BB1.GRATicule.AXIS.Y.STATe = Var
Var = SCPI.DISPlay.BB1.GRATicule.AXIS.Y.STATe
```

Related Objects

SCPI.DISPlay.BB(Ch).GRATicule.AXIS.Y.RELative

Equivalent Key

BB Menu: **Display** > **Y # of Digits**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:GRATicule:AXIS:Y:STATe {OFF|SHORt|MIDDle|LONG}

:DISPlay:BB[1-1]:GRATicule:AXIS:Y:STATe?

Query Response

{OFF|SHORt|MIDDle|LONG} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:GRATicule:AXIS:Y:STATe OFF"
20 OUTPUT 717;":DISPlay:BB1:GRATicule:AXIS:Y:STATe ?"
30 ENTER 717;A$
```

SCPI.DISPlay.BB(Ch).LABel.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.BB(Ch).LABel.DATA = *Value*

Value = SCPI.DISPlay.BB(Ch).LABel.DATA

Description

This command sets/gets Window Title Label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Window Title Label
Data Type	Character string type (String)
Range	254 chars
Preset Value	""
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
SCPI.DISPlay.BB1.LABel.DATA = Var
Var = SCPI.DISPlay.BB1.LABel.DATA
```

Related Objects

SCPI.DISPlay.BB(Ch).LABel.STATe

Equivalent Key

BB Menu: **Display** > **Edit Title Label**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:LABel:DATA

:DISPlay:BB[1-1]:LABel:DATA?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:LABel:DATA Sample_Text"
20 OUTPUT 717;":DISPlay:BB1:LABel:DATA ?"
30 ENTER 717;A$
```

SCPI.DISPlay.BB(Ch).LABel.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.BB(Ch).LABel.STATe = *Value*

Value = SCPI.DISPlay.BB(Ch).LABel.STATe

Description

This command sets/gets Show/Hide Window Title Label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/Hide Window Title Label
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.BB1.LABel.STATe = Var
Var = SCPI.DISPlay.BB1.LABel.STATe
```

Related Objects

SCPI.DISPlay.BB(Ch).LABel.DATA

Equivalent Key

BB Menu: **Display** > **Title Label**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:LABel:STATe {ON|OFF|1|0}

:DISPlay:BB[1-1]:LABel:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:LABel:STATe 1"
20 OUTPUT 717;":DISPlay:BB1:LABel:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.BB(Ch).LIMit.FSIGn

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.BB(Ch).LIMit.FSIGn = *Value*

Value = SCPI.DISPlay.BB(Ch).LIMit.FSIGn

Description

This command sets/gets show/hide fail sign, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	show/hide fail sign
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.BB1.LIMit.FSIGn = Var
Var = SCPI.DISPlay.BB1.LIMit.FSIGn
```

Equivalent Key

BB Menu: **Display** > **Limit Test** > **Fail Sign**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:LIMit:FSIGn {ON|OFF|1|0}

:DISPlay:BB[1-1]:LIMit:FSIGn?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:LIMit:FSIGn 1"
20 OUTPUT 717;":DISPlay:BB1:LIMit:FSIGn ?"
30 ENTER 717;A
```

SCPI.DISPlay.BB(Ch).MAXimize

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.BB(Ch).MAXimize = *Value*

Value = SCPI.DISPlay.BB(Ch).MAXimize

Description

This command sets/gets maximize active trace, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Maximize active trace
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.BB1.MAXimize = Var
Var = SCPI.DISPlay.BB1.MAXimize
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:MAXimize {ON|OFF|1|0}

:DISPlay:BB[1-1]:MAXimize?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:MAXimize 1"
20 OUTPUT 717;":DISPlay:BB1:MAXimize ?"
30 ENTER 717;A
```

SCPI.DISPlay.BB(Ch).STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.BB(Ch).STATe = *Value*

Value = SCPI.DISPlay.BB(Ch).STATe

Description

This command sets/gets measurement display on/off. At least one measurement window must be turned on., for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Measurement display (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	60, "Failed to hide window" 100, "Option not installed"

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.BB1.STATe = Var
Var = SCPI.DISPlay.BB1.STATe
```

Equivalent Key

PN Menu: **Measurement View** > **Show Window** > **Baseband Noise**

SP Menu: **Measurement View** > **Show Window** > **Baseband Noise**

FP Menu: **Measurement View** > **Show Window** > **Baseband Noise**

TR Menu: **Measurement View** > **Show Window** > **Baseband Noise**

AM Menu: **Measurement View** > **Show Window** > **Baseband Noise**

BB Menu: **Measurement View** > **Show Window** > **Baseband Noise**

USER Menu: **Measurement View** > **Show Window** > **Baseband Noise**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:STATe {ON|OFF|1|0}

:DISPlay:BB[1-1]:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:STATe 1"
20 OUTPUT 717;":DISPlay:BB1:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.BB(Ch).TABLe.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.BB(Ch).TABLe.STATe = *Value*

Value = SCPI.DISPlay.BB(Ch).TABLe.STATe

Description

This command sets/gets show/hide text area, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/hide text area
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.BB1.TABLe.STATe = Var
Var = SCPI.DISPlay.BB1.TABLe.STATe
```

Related Objects

Equivalent Key

BB Menu: **Marker** > **Marker List**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:TABLe[:STATe] {ON|OFF|1|0}

:DISPlay:BB[1-1]:TABLe[:STATe]?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:TABLe[:STATe] 1"
20 OUTPUT 717;":DISPlay:BB1:TABLe[:STATe] ?"
30 ENTER 717;A
```

SCPI.DISPlay.BB(Ch).TRACe(Tr).LABel.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.BB(Ch).TRACe(Tr).LABel.DATA = *Value*

Value = SCPI.DISPlay.BB(Ch).TRACe(Tr).LABel.DATA

Description

This command sets/gets Trace Title Label, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace Title Label
Data Type	Character string type (String)
Range	254 chars
Preset Value	"Baseband"
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
SCPI.DISPlay.BB1.TRACe1.LABel.DATA = Var
Var = SCPI.DISPlay.BB1.TRACe1.LABel.DATA
```

Equivalent Key

BB Menu: **Trace View** > **Trace Label**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:TRACe[1-1]:LABel:DATA

:DISPlay:BB[1-1]:TRACe[1-1]:LABel:DATA?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:TRACe1:LABel:DATA Sample_Text"
20 OUTPUT 717;":DISPlay:BB1:TRACe1:LABel:DATA ?"
30 ENTER 717;A$
```

SCPI.DISPlay.BB(Ch).TRACe(Tr).LIMit.LINE

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.BB(Ch).TRACe(Tr).LIMit.LINE = *Value*

Value = SCPI.DISPlay.BB(Ch).TRACe(Tr).LIMit.LINE

Description

This command sets/gets show/hide limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/hide limit line
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.BB1.TRACe1.LIMit.LINE = Var
Var = SCPI.DISPlay.BB1.TRACe1.LIMit.LINE
```

Equivalent Key

BB Menu: **Display** > **Limit Test** > **Limit Line**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:TRACe[1-1]:LIMit:LINE {ON|OFF|1|0}

:DISPlay:BB[1-1]:TRACe[1-1]:LIMit:LINE?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:TRACe1:LIMit:LINE 1"
20 OUTPUT 717;":DISPlay:BB1:TRACe1:LIMit:LINE ?"
30 ENTER 717;A
```

SCPI.DISPlay.BB(Ch).TRACe(Tr).MODE

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.BB(Ch).TRACe(Tr).MODE = *Value*

Value = SCPI.DISPlay.BB(Ch).TRACe(Tr).MODE

Description

This command sets/gets show data and/or memory trace, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show data and/or memory trace
Data Type	Character string type (String)
Range	OFF|DATA|MEMory|BOTH
Preset Value	DATA
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.DISPlay.BB1.TRACe1.MODE = Var
Var = SCPI.DISPlay.BB1.TRACe1.MODE
```

Equivalent Key

BB Menu: **Trace View** > **Display Trace**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:TRACe[1-1]:MODE {OFF|DATA|MEMory|BOTH}

:DISPlay:BB[1-1]:TRACe[1-1]:MODE?

Query Response

{OFF|DATA|MEMory|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:TRACe1:MODE OFF"
20 OUTPUT 717;":DISPlay:BB1:TRACe1:MODE ?"
30 ENTER 717;A$
```

SCPI.DISPlay.BB(Ch).TRACe(Tr).PERSistence.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.BB(Ch).TRACe(Tr).PERSistence.CLEar

Description

This command clears stored traces, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.DISPlay.BB1.TRACe1.PERSistence.CLEar

Related Objects

SCPI.DISPlay.BB(Ch).TRACe(Tr).PERSistence.STATe

Equivalent Key

BB Menu: **Trace View** > **Persistence** > **Clear Persistent Data**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:TRACe[1-1]:PERSistence:CLEar

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:TRACe1:PERSistence:CLEar"
```

SCPI.DISPlay.BB(Ch).TRACe(Tr).PERSistence.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.BB(Ch).TRACe(Tr).PERSistence.STATe = *Value*

Value = SCPI.DISPlay.BB(Ch).TRACe(Tr).PERSistence.STATe

Description

This command sets/gets trace storage mode, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace storage mode
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.BB1.TRACe1.PERSistence.STATe = Var
Var = SCPI.DISPlay.BB1.TRACe1.PERSistence.STATe
```

Related Objects

SCPI.DISPlay.BB(Ch).TRACe(Tr).PERSistence.CLEar

Equivalent Key

BB Menu: **Trace View** > **Persistence** > **Persistence Mode**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:TRACe[1-1]:PERSistence:STATe {ON|OFF|1|0}

:DISPlay:BB[1-1]:TRACe[1-1]:PERSistence:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:TRACe1:PERSistence:STATe 1"
20 OUTPUT 717;":DISPlay:BB1:TRACe1:PERSistence:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.BB(Ch).TRACe(Tr).X.SCALe.AUTO

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.BB(Ch).TRACe(Tr).X.SCALe.AUTO = *Value*

Value = SCPI.DISPlay.BB(Ch).TRACe(Tr).X.SCALe.AUTO

Description

This command sets/gets autoscale setting for X axis, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	autoscale setting for X axis
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.BB1.TRACe1.X.SCALe.AUTO = Var
Var = SCPI.DISPlay.BB1.TRACe1.X.SCALe.AUTO
```

Related Objects

SCPI.DISPlay.BB(Ch).TRACe(Tr).X.SCALe.AUTO

SCPI.DISPlay.BB(Ch).TRACe(Tr).X.SCALe.LEFT

SCPI.DISPlay.BB(Ch).TRACe(Tr).X.SCALe.RIGHt

Equivalent Key

BB Menu: **Scale** > **X Axis** > **Auto**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:TRACe[1-1]:X[:SCALe]:AUTO {ON|OFF|1|0}

:DISPlay:BB[1-1]:TRACe[1-1]:X[:SCALe]:AUTO?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:TRACe1:X[:SCALe]:AUTO 1"
20 OUTPUT 717;":DISPlay:BB1:TRACe1:X[:SCALe]:AUTO ?"
30 ENTER 717;A
```

SCPI.DISPlay.BB(Ch).TRACe(Tr).X.SCALe.LEFT

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.BB(Ch).TRACe(Tr).X.SCALe.LEFT = *Value*

Value = SCPI.DISPlay.BB(Ch).TRACe(Tr).X.SCALe.LEFT

Description

This command sets/gets left setting for X axis, for the selected trace *Tr* of the selected channel *Ch*.

NOTE This command is available only when the automatic setting of the X-axis display range (set with SCPI.DISPlay.BB(Ch).TRACe(Tr).X.SCALe.AUTO) is set to OFF.

Variable

Parameter	*Value*
Description	Left setting for X axis
Data Type	Double precision floating point type (Double)
Range	1 ~ 99.9999999M
Preset Value	1k
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.DISPlay.BB1.TRACe1.X.SCALe.LEFT = Var
Var = SCPI.DISPlay.BB1.TRACe1.X.SCALe.LEFT
```

Related Objects

SCPI.DISPlay.BB(Ch).TRACe(Tr).X.SCALe.AUTO

SCPI.DISPlay.BB(Ch).TRACe(Tr).X.SCALe.RIGHt

Equivalent Key

BB Menu: **Scale** > **X Axis** > **Left**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:TRACe[1-1]:X[:SCALe]:LEFT {1 ~ 99.9999999M}

:DISPlay:BB[1-1]:TRACe[1-1]:X[:SCALe]:LEFT?

Query Response

{1 ~ 99.9999999M} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:TRACe1:X[:SCALe]:LEFT 0"
20 OUTPUT 717;":DISPlay:BB1:TRACe1:X[:SCALe]:LEFT ?"
30 ENTER 717;A
```

SCPI.DISPlay.BB(Ch).TRACe(Tr).X.SCALe.RIGHt

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.BB(Ch).TRACe(Tr).X.SCALe.RIGHt = *Value*

Value = SCPI.DISPlay.BB(Ch).TRACe(Tr).X.SCALe.RIGHt

Description

This command sets/gets right setting for X axis, for the selected trace *Tr* of the selected channel *Ch*.

NOTE This command is available only when the automatic setting of the X-axis display range (set with SCPI.DISPlay.BB(Ch).TRACe(Tr).X.SCALe.AUTO) is set to OFF

Variable

Parameter	*Value*
Description	Right setting for X axis
Data Type	Double precision floating point type (Double)
Range	1.1 ~ 100M
Preset Value	10M
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.DISPlay.BB1.TRACe1.X.SCALe.RIGHt = Var
Var = SCPI.DISPlay.BB1.TRACe1.X.SCALe.RIGHt
```

Related Objects

SCPI.DISPlay.BB(Ch).TRACe(Tr).X.SCALe.AUTO

SCPI.DISPlay.BB(Ch).TRACe(Tr).X.SCALe.LEFT

Equivalent Key

BB Menu: **Scale** > **X Axis** > **Right**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:TRACe[1-1]:X[:SCALe]:RIGHt {1.1 ~ 100M}

:DISPlay:BB[1-1]:TRACe[1-1]:X[:SCALe]:RIGHt?

Query Response

{1.1 ~ 100M} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:TRACe1:X[:SCALe]:RIGHt 0"
20 OUTPUT 717;":DISPlay:BB1:TRACe1:X[:SCALe]:RIGHt ?"
30 ENTER 717;A
```

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.AUTO

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.AUTO

Description

This command sets auto scale for Y axis, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.DISPlay.BB1.TRACe1.Y.SCALe.AUTO

Related Objects

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.BOTTom

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.RLEVel

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.RPOSition

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.TOP

Equivalent Key

BB Menu: **Scale** > **Auto Scale**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:TRACe[1-1]:Y[:SCALe]:AUTO

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:TRACe1:Y[:SCALe]:AUTO"
```

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.BOTTom

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.BOTTom = *Value*

Value = SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.BOTTom

Description

This command sets/gets scale bottom level, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale bottom level
Data Type	Double precision floating point type (Double)
Range	-350G ~ 70G
Preset Value	-200
Unit	V/Hz
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -350000000000
SCPI.DISPlay.BB1.TRACe1.Y.SCALe.BOTTom = Var
Var = SCPI.DISPlay.BB1.TRACe1.Y.SCALe.BOTTom
```

Related Objects

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.RLEVel

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.RPOSition

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.TOP

Equivalent Key

BB Menu: **Scale** > **Bottom**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:TRACe[1-1]:Y[:SCALe]:BOTTom {-350G ~ 70G}

:DISPlay:BB[1-1]:TRACe[1-1]:Y[:SCALe]:BOTTom?

Query Response

{-350G ~ 70G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:TRACe1:Y[:SCALe]:BOTTom -350000000000 "
20 OUTPUT 717;":DISPlay:BB1:TRACe1:Y[:SCALe]:BOTTom ?"
30 ENTER 717;A
```

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.PDIVision

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.PDIVision = *Value*

Value = SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.PDIVision

Description

This command sets/gets scale per division, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale per division
Data Type	Double precision floating point type (Double)
Range	1a ~ 10G
Preset Value	10
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0.000000000001
SCPI.DISPlay.BB1.TRACe1.Y.SCALe.PDIVision = Var
Var = SCPI.DISPlay.BB1.TRACe1.Y.SCALe.PDIVision
```

Related Objects

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.BOTTom

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.RLEVel

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.RPOSition

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.TOP

Equivalent Key

BB Menu: **Scale** > **Scale/Div**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:TRACe[1-1]:Y[:SCALe]:PDIVision {1a ~ 10G}

:DISPlay:BB[1-1]:TRACe[1-1]:Y[:SCALe]:PDIVision?

Query Response

{1a ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:TRACe1:Y[:SCALe]:PDIVision 0.000000000001"
20 OUTPUT 717;":DISPlay:BB1:TRACe1:Y[:SCALe]:PDIVision ?"
30 ENTER 717;A
```

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.RLEVel

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.RLEVel = *Value*

Value = SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.RLEVel

Description

This command sets/gets scale reference level, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale reference level
Data Type	Double precision floating point type (Double)
Range	-50G ~ 50G
Preset Value	-80
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -50000000000
SCPI.DISPlay.BB1.TRACe1.Y.SCALe.RLEVel = Var
Var = SCPI.DISPlay.BB1.TRACe1.Y.SCALe.RLEVel
```

Related Objects

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.BOTTom

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.RPOSition

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.TOP

Equivalent Key

BB Menu: **Scale** > **Reference Value**

BB Menu: **Scale** > **Marker** > **Reference**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:TRACe[1-1]:Y[:SCALe]:RLEVel {-50G ~ 50G}

:DISPlay:BB[1-1]:TRACe[1-1]:Y[:SCALe]:RLEVel?

Query Response

{-50G ~ 50G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:TRACe1:Y[:SCALe]:RLEVel -50000000000"
20 OUTPUT 717;":DISPlay:BB1:TRACe1:Y[:SCALe]:RLEVel ?"
30 ENTER 717;A
```

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.RPOSition

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.RPOSition = *Value*

Value = SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.RPOSition

Description

This command sets/gets scale reference position, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale reference position
Data Type	Long integer type (Long)
Range	0 ~ 30
Preset Value	12
Unit	Div
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.DISPlay.BB1.TRACe1.Y.SCALe.RPOSition = Var
Var = SCPI.DISPlay.BB1.TRACe1.Y.SCALe.RPOSition
```

Related Objects

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.BOTTom

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.RLEVel

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.TOP

Equivalent Key

BB Menu: **Scale** > **Reference Position**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:TRACe[1-1]:Y[:SCALe]:RPOSition {0 ~ 30}

:DISPlay:BB[1-1]:TRACe[1-1]:Y[:SCALe]:RPOSition?

Query Response

{0 ~ 30} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:TRACe1:Y[:SCALe]:RPOSition 0 "
20 OUTPUT 717;":DISPlay:BB1:TRACe1:Y[:SCALe]:RPOSition ?"
30 ENTER 717;A
```

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.TOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.TOP = *Value*

Value = SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.TOP

Description

This command sets/gets scale top level, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale top level
Data Type	Double precision floating point type (Double)
Range	-50G ~ 350G
Preset Value	-80
Unit	V/Hz
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -50000000000
SCPI.DISPlay.BB1.TRACe1.Y.SCALe.TOP = Var
Var = SCPI.DISPlay.BB1.TRACe1.Y.SCALe.TOP
```

Related Objects

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.BOTTom

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.RLEVel

SCPI.DISPlay.BB(Ch).TRACe(Tr).Y.SCALe.RPOSition

Equivalent Key

BB Menu: **Scale** > **Top**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:TRACe[1-1]:Y[:SCALe]:TOP {-50G ~ 350G}

:DISPlay:BB[1-1]:TRACe[1-1]:Y[:SCALe]:TOP?

Query Response

{-50G ~ 350G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:TRACe1:Y[:SCALe]:TOP -50000000000 "
20 OUTPUT 717;":DISPlay:BB1:TRACe1:Y[:SCALe]:TOP ?"
30 ENTER 717;A
```

SCPI.DISPlay.BB(Ch).Y.SCALe.DIVisions

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.BB(Ch).Y.SCALe.DIVisions = *Value*

Value = SCPI.DISPlay.BB(Ch).Y.SCALe.DIVisions

Description

This command sets/gets number of Y division, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Number of Y division
Data Type	Long integer type (Long)
Range	4 ~ 30
Preset Value	12
Unit	-
Resolution	2
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 4
SCPI.DISPlay.BB1.Y.SCALe.DIVisions = Var
Var = SCPI.DISPlay.BB1.Y.SCALe.DIVisions
```

Equivalent Key

BB Menu: **Scale** > **Divisions**

Equivalent SCPI Command

Syntax

:DISPlay:BB[1-1]:Y[:SCALe]:DIVisions {4 ~ 30}

:DISPlay:BB[1-1]:Y[:SCALe]:DIVisions?

Query Response

{4 ~ 30} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:BB1:Y[:SCALe]:DIVisions 4 "
20 OUTPUT 717;":DISPlay:BB1:Y[:SCALe]:DIVisions ?"
30 ENTER 717;A
```

SCPI.DISPlay.CLOCk

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.CLOCk = *Value*

Value = SCPI.DISPlay.CLOCk

Description

This command Show/Hide clock display.

Variable

Parameter	*Value*
Description	Show/Hide clock display
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.CLOCk = Var
Var = SCPI.DISPlay.CLOCk
```

Equivalent Key

PN Menu: **System** > **Misc Setup** > **Clock Setup** > **Show Clock**

SP Menu: **System** > **Misc Setup** > **Clock Setup** > **Show Clock**

FP Menu: **System** > **Misc Setup** > **Clock Setup** > **Show Clock**

TR Menu: **System** > **Misc Setup** > **Clock Setup** > **Show Clock**

AM Menu: **System** > **Misc Setup** > **Clock Setup** > **Show Clock**

BB Menu: **System** > **Misc Setup** > **Clock Setup** > **Show Clock**

USER Menu: **System** > **Misc Setup** > **Clock Setup** > **Show Clock**

Equivalent SCPI Command

Syntax

:DISPlay:CLOCk {ON|OFF|1|0}

:DISPlay:CLOCk?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:CLOCk 1"
20 OUTPUT 717;":DISPlay:CLOCk ?"
30 ENTER 717;A
```

SCPI.DISPlay.COLor(1-2).BACK.VALue[_Q] red, green, blue

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.COLor(1-2).BACK.VALue[_Q] red, green, blue = *Value*

Value = SCPI.DISPlay.COLor(1-2).BACK.VALue[_Q] red, green, blue

Description

This command sets/gets Background color.

Variable

Parameter	*red*
Description	Red color value in background color
Data Type	Long integer type (Long)
Range	0 ~ 255
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Parameter	*green*
Description	Green color value in background color
Data Type	Long integer type (Long)
Range	0 ~ 255
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*blue*
Description	Blue color value in background color
Data Type	Long integer type (Long)
Range	0 ~ 255
Preset Value	0
Unit	-

Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
SCPI.DISPlay.COLor2.BACK.VALue[_Q] red, green, blue = Var
Var = SCPI.DISPlay.COLor2.BACK.VALue[_Q] red, green, blue
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:COLor[1-2]:BACK[:VALue] {0 ~ 255 0 ~ 255 0 ~ 255}

:DISPlay:COLor[1-2]:BACK[:VALue]?

Query Response

{0 ~ 255 0 ~ 255 0 ~ 255} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:COLor2:BACK[:VALue] 1"
20 OUTPUT 717;":DISPlay:COLor2:BACK[:VALue] ?"
30 ENTER 717;A
```

SCPI.DISPlay.COLor(1-2).GRATicule(1-2).VALue[_Q] red, green, blue

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.COLor(1-2).GRATicule(1-2).VALue[_Q] red, green, blue = *Value*

Value = SCPI.DISPlay.COLor(1-2).GRATicule(1-2).VALue[_Q] red, green, blue

Description

This command sets/gets Graticule color.

Variable

Parameter	*red*
Description	Red color value in Graticule color
Data Type	Long integer type (Long)
Range	0 ~ 255
Preset Value	127
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Parameter	*green*
Description	Green color value in Graticule color
Data Type	Long integer type (Long)
Range	0 ~ 255
Preset Value	127
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*blue*
Description	Blue color value in Graticule color
Data Type	Long integer type (Long)
Range	0 ~ 255
Preset Value	127
Unit	-

Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
SCPI.DISPlay.COLor2.GRATicule2.VALue[_Q] red, green, blue = Var
Var = SCPI.DISPlay.COLor2.GRATicule2.VALue[_Q] red, green, blue
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:COLor[1-2]:GRATicule[1-2][:VALue] {0 ~ 255 0 ~ 255 0 ~ 255}

:DISPlay:COLor[1-2]:GRATicule[1-2][:VALue]?

Query Response

{0 ~ 255 0 ~ 255 0 ~ 255} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:COLor2:GRATicule2[:VALue] 1"
20 OUTPUT 717;":DISPlay:COLor2:GRATicule2[:VALue] ?"
30 ENTER 717;A
```

SCPI.DISPlay.COLor(1-2).LIMit(1-2).VALue[_Q] red, green, blue

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.COLor(1-2).LIMit(1-2).VALue[_Q] red, green, blue = *Value*

Value = SCPI.DISPlay.COLor(1-2).LIMit(1-2).VALue[_Q] red, green, blue

Description

This command sets/gets Limit color (1:Fail color 2:Line color).

Variable

Parameter	*red*
Description	Red color value in Limit color
Data Type	Long integer type (Long)
Range	0 ~ 255
Preset Value	255
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Parameter	*green*
Description	Green color value in Limit color
Data Type	Long integer type (Long)
Range	0 ~ 255
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*blue*
Description	Blue color value in Limit color
Data Type	Long integer type (Long)
Range	0 ~ 255
Preset Value	0
Unit	-

Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
SCPI.DISPlay.COLor2.LIMit2.VALue[_Q] red, green, blue = Var
Var = SCPI.DISPlay.COLor2.LIMit2.VALue[_Q] red, green, blue
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:COLor[1-2]:LIMit[1-2][:VALue] {0 ~ 255 0 ~ 255 0 ~ 255}

:DISPlay:COLor[1-2]:LIMit[1-2][:VALue]?

Query Response

{0 ~ 255 0 ~ 255 0 ~ 255} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:COLor2:LIMit2[:VALue] 1"
20 OUTPUT 717;":DISPlay:COLor2:LIMit2[:VALue] ?"
30 ENTER 717;A
```

SCPI.DISPlay.COLor(1-2).RESet

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.COLor(1-2).RESet

Description

This command resets the colors to their default (factory) value.

Examples

SCPI.DISPlay.COLor2.RESet

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:COLor[1-2]:RESet

Example of use

10 OUTPUT 717;":DISPlay:COLor2:RESet"

SCPI.DISPlay.COLor(1-2).TRACe(Tr).DATA.VALue[_Q] red, green, blue

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.COLor(1-2).TRACe(Tr).DATA.VALue[_Q] red, green, blue = *Value*

Value = SCPI.DISPlay.COLor(1-2).TRACe(Tr).DATA.VALue[_Q] red, green, blue

Description

This command sets/gets Trace data color.

Variable

Parameter	*red*
Description	Red color value in trace data color
Data Type	Long integer type (Long)
Range	0 ~ 255
Preset Value	255
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Parameter	*green*
Description	Green color value in trace data color
Data Type	Long integer type (Long)
Range	0 ~ 255
Preset Value	255
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*blue*
Description	Blue color value in trace data color
Data Type	Long integer type (Long)
Range	0 ~ 255
Preset Value	0
Unit	-

Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
SCPI.DISPlay.COLor2.TRACe8.DATA.VALue[_Q] red, green, blue = Var
Var = SCPI.DISPlay.COLor2.TRACe8.DATA.VALue[_Q] red, green, blue
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:COLor[1-2]:TRACe[1-8]:DATA[:VALue] {0 ~ 255 0 ~ 255 0 ~ 255}

:DISPlay:COLor[1-2]:TRACe[1-8]:DATA[:VALue]?

Query Response

{0 ~ 255 0 ~ 255 0 ~ 255} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:COLor2:TRACe8:DATA[:VALue] 1"
20 OUTPUT 717;":DISPlay:COLor2:TRACe8:DATA[:VALue] ?"
```

SCPI.DISPlay.COLor(1-2).TRACe(Tr).MEMory.VALue[_Q] red, green, blue

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.COLor(1-2).TRACe(Tr).MEMory.VALue[_Q] red, green, blue = *Value*

Value = SCPI.DISPlay.COLor(1-2).TRACe(Tr).MEMory.VALue[_Q] red, green, blue

Description

This command sets/gets Trace memory color.

Variable

Parameter	*red*
Description	Red color value in trace memory color
Data Type	Long integer type (Long)
Range	0 ~ 255
Preset Value	127
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Parameter	*green*
Description	Green color value in trace memory color
Data Type	Long integer type (Long)
Range	0 ~ 255
Preset Value	127
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*blue*
Description	Blue color value in trace memory color
Data Type	Long integer type (Long)
Range	0 ~ 255

Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
Var= 1
SCPI.DISPlay.COLor2.TRACe8.MEMory.VALue[_Q] red, green, blue = Var
Var = SCPI.DISPlay.COLor2.TRACe8.MEMory.VALue[_Q] red, green, blue
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:COLor[1-2]:TRACe[1-8]:MEMory[:VALue] {0 ~ 255 0 ~ 255 0 ~ 255}

:DISPlay:COLor[1-2]:TRACe[1-8]:MEMory[:VALue]?

Query Response

{0 ~ 255 0 ~ 255 0 ~ 255} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:COLor2:TRACe8:MEMory[:VALue] 1"
20 OUTPUT 717;":DISPlay:COLor2:TRACe8:MEMory[:VALue] ?"
```

SCPI.DISPlay.ECHO.ADD

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.ECHO.ADD = *Value*

Description

This command sets add text in echo window except LineFeed(0x0a, vbLF) character. Other non-printable characters are converted to a space character. The excess portion of SCPI.DISPlay.ECHO.DATA string will be silently ignored.

Variable

Parameter	*Value*
Description	Text in echo window
Data Type	Character string type (String)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as String
SCPI.DISPlay.ECHO.ADD = Var
```

Related Objects

SCPI.DISPlay.ECHO.CLEar

SCPI.DISPlay.ECHO.DATA

SCPI.DISPlay.ECHO.FSIZe

SCPI.DISPlay.ECHO.STATe

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:ECHO:ADD

Query Response

{String} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:ECHO:ADD Sample_Text"
20 ENTER 717;A$
```

SCPI.DISPlay.ECHO.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.ECHO.CLEar

Description

This command clears echo window.

Examples

SCPI.DISPlay.ECHO.CLEar

Related Objects

SCPI.DISPlay.ECHO.ADD

SCPI.DISPlay.ECHO.DATA

SCPI.DISPlay.ECHO.FSIZe

SCPI.DISPlay.ECHO.STATe

Equivalent Key

PN Menu: **Macro Setup** > **Echo Window Menu** > **Clear Echo**

SP Menu: **Macro Setup** > **Echo Window Menu** > **Clear Echo**

FP Menu: **Macro Setup** > **Echo Window Menu** > **Clear Echo**

TR Menu: **Macro Setup** > **Echo Window Menu** > **Clear Echo**

AM Menu: **Macro Setup** > **Echo Window Menu** > **Clear Echo**

BB Menu: **Macro Setup** > **Echo Window Menu** > **Clear Echo**

USER Menu: **Macro Setup** > **Echo Window Menu** > **Clear Echo**

Equivalent SCPI Command

Syntax

:DISPlay:ECHO:CLEar

Example of use

```
10 OUTPUT 717;":DISPlay:ECHO:CLEar"
```

SCPI.DISPlay.ECHO.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.ECHO.DATA = *Value*

Value = SCPI.DISPlay.ECHO.DATA

Description

This command sets/gets text in echo window except LineFeed(0x0a, vbLF) character. Other non-printable characters are converted to a space charcter..

Variable

Parameter	*Value*
Description	Text in echo window
Data Type	Character string type (String)
Range	-
Preset Value	""
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as String
SCPI.DISPlay.ECHO.DATA = Var
Var = SCPI.DISPlay.ECHO.DATA
```

Related Objects

SCPI.DISPlay.ECHO.ADD

SCPI.DISPlay.ECHO.CLEar

SCPI.DISPlay.ECHO.FSIZe

SCPI.DISPlay.ECHO.STATe

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:ECHO:DATA

:DISPlay:ECHO:DATA?

Query Response

{String} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:ECHO:DATA Sample_Text"
20 OUTPUT 717;":DISPlay:ECHO:DATA ?"
30 ENTER 717;A$
```

SCPI.DISPlay.ECHO.FSIZe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.ECHO.FSIZe = *Value*

Value = SCPI.DISPlay.ECHO.FSIZe

Description

This command sets/gets font size in echo window.

Variable

Parameter	*Value*
Description	Font size in echo window
Data Type	Long integer type (Long)
Range	9|10|11|12|14|16|18|20|22|24|26|28|36|48|72|96|112
Preset Value	11
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 9
SCPI.DISPlay.ECHO.FSIZe = Var
Var = SCPI.DISPlay.ECHO.FSIZe
```

Related Objects

SCPI.DISPlay.ECHO.ADD

SCPI.DISPlay.ECHO.CLEar

SCPI.DISPlay.ECHO.DATA

SCPI.DISPlay.ECHO.STATe

Equivalent Key

PN Menu: **Macro Setup** > **Echo Window Menu** > **Echo Font Size**

SP Menu: **Macro Setup** > **Echo Window Menu** > **Echo Font Size**

FP Menu: **Macro Setup** > **Echo Window Menu** > **Echo Font Size**

TR Menu: **Macro Setup** > **Echo Window Menu** > **Echo Font Size**

AM Menu: **Macro Setup** > **Echo Window Menu** > **Echo Font Size**

BB Menu: **Macro Setup** > **Echo Window Menu** > **Echo Font Size**

USER Menu: **Macro Setup** > **Echo Window Menu** > **Echo Font Size**

Equivalent SCPI Command

Syntax

:DISPlay:ECHO:FSIZe
{9|10|11|12|14|16|18|20|22|24|26|28|36|48|72|96|112}

:DISPlay:ECHO:FSIZe?

Query Response

{9|10|11|12|14|16|18|20|22|24|26|28|36|48|72|96|112} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:ECHO:FSIZe 9"
20 OUTPUT 717;":DISPlay:ECHO:FSIZe ?"
30 ENTER 717;A
```

SCPI.DISPlay.ECHO.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.ECHO.STATe = *Value*

Value = SCPI.DISPlay.ECHO.STATe

Description

This command Show/Hide echo window.

Variable

Parameter	*Value*
Description	Show/Hide echo window
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Boolean
Var= 1
```

```
SCPI.DISPlay.ECHO.STATe = Var
Var = SCPI.DISPlay.ECHO.STATe
```

Related Objects

SCPI.DISPlay.ECHO.ADD

SCPI.DISPlay.ECHO.CLEar

SCPI.DISPlay.ECHO.DATA

SCPI.DISPlay.ECHO.FSIZe

Equivalent Key

PN Menu: **Macro Setup** > **Echo Window Menu** > **Echo Window**

SP Menu: **Macro Setup** > **Echo Window Menu** > **Echo Window**

FP Menu: **Macro Setup** > **Echo Window Menu** > **Echo Window**

TR Menu: **Macro Setup** > **Echo Window Menu** > **Echo Window**

AM Menu: **Macro Setup** > **Echo Window Menu** > **Echo Window**

BB Menu: **Macro Setup** > **Echo Window Menu** > **Echo Window**

USER Menu: **Macro Setup** > **Echo Window Menu** > **Echo Window**

Equivalent SCPI Command

Syntax

:DISPlay:ECHO:STATe {ON|OFF|1|0}

:DISPlay:ECHO:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:ECHO:STATe 1"
20 OUTPUT 717;":DISPlay:ECHO:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.ENABle

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.ENABle = *Value*

Value = SCPI.DISPlay.ENABle

Description

This command Enables/disables trace update.

Variable

Parameter	*Value*
Description	Enables/disables trace update
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.ENABle = Var
Var = SCPI.DISPlay.ENABle
```

Equivalent Key

PN Menu: **Display** > **Update**

SP Menu: **Display** > **Update**

FP Menu: **Display** > **Update**

TR Menu: **Display** > **Update**

AM Menu: **Display** > **Update**

BB Menu: **Display** > **Update**

PS Menu: **Display** > **Update**

USER Menu: **Display** > **Update**

Equivalent SCPI Command

Syntax

:DISPlay:ENABle {ON|OFF|1|0}

:DISPlay:ENABle?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:ENABle 1"
20 OUTPUT 717;":DISPlay:ENABle ?"
30 ENTER 717;A
```

SCPI.DISPlay.FP(Ch).ALLTrace.PERSistence.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.FP(Ch).ALLTrace.PERSistence.CLEar

Description

This command clears all stored traces, for the selected channel *Ch*.

Examples

SCPI.DISPlay.FP1.ALLTrace.PERSistence.CLEar

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:ALLTrace:PERSistence:CLEar

Example of use

10 OUTPUT 717;":DISPlay:FP1:ALLTrace:PERSistence:CLEar"

SCPI.DISPlay.FP(Ch).ALLTrace.Y.SCALe.AUTO

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.FP(Ch).ALLTrace.Y.SCALe.AUTO

Description

This command sets auto scale all, for the selected channel *Ch*.

Examples

SCPI.DISPlay.FP1.ALLTrace.Y.SCALe.AUTO

Equivalent Key

FP Menu: **Scale** > **Auto Scale All**

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:ALLTrace:Y:SCALe:AUTO

Example of use

10 OUTPUT 717;":DISPlay:FP1:ALLTrace:Y:SCALe:AUTO"

SCPI.DISPlay.FP(Ch).ANNotation.MARKer.POSition

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.FP(Ch).ANNotation.MARKer.POSition = *Value*

Value = SCPI.DISPlay.FP(Ch).ANNotation.MARKer.POSition

Description

This command sets/gets Marker annotation position, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker annotation position
Data Type	Character string type (String)
Range	LEFT|RIGHt
Preset Value	LEFT
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "LEFT"
SCPI.DISPlay.FP1.ANNotation.MARKer.POSition = Var
Var = SCPI.DISPlay.FP1.ANNotation.MARKer.POSition
```

Related Objects

SCPI.DISPlay.FP(Ch).ANNotation.MEASurement.STATe

Equivalent Key

FP Menu: **Display** > **Marker Information**

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:ANNotation:MARKer:POSition {LEFT|RIGHt}

:DISPlay:FP[1-1]:ANNotation:MARKer:POSition?

Query Response

{LEFT|RIGHt} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:FP1:ANNotation:MARKer:POSition LEFT"
20 OUTPUT 717;":DISPlay:FP1:ANNotation:MARKer:POSition ?"
30 ENTER 717;A$
```

SCPI.DISPlay.FP(Ch).ANNotation.MEASurement.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.FP(Ch).ANNotation.MEASurement.STATe = *Value*

Value = SCPI.DISPlay.FP(Ch).ANNotation.MEASurement.STATe

Description

This command sets/gets measurement setting, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Measurement setting
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.FP1.ANNotation.MEASurement.STATe = Var
Var = SCPI.DISPlay.FP1.ANNotation.MEASurement.STATe
```

Related Objects

SCPI.DISPlay.FP(Ch).ANNotation.MARKer.POSition

Equivalent Key

FP Menu: **Display** > **Meas Condition**

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:ANNotation:MEASurement:STATe {ON|OFF|1|0}

:DISPlay:FP[1-1]:ANNotation:MEASurement:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:FP1:ANNotation:MEASurement:STATe 1"
20 OUTPUT 717;":DISPlay:FP1:ANNotation:MEASurement:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.FP(Ch).GRATicule.AXIS.Y.RELative

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.FP(Ch).GRATicule.AXIS.Y.RELative = *Value*

Value = SCPI.DISPlay.FP(Ch).GRATicule.AXIS.Y.RELative

Description

This command sets/gets force graticule label notation relative. If OFF, absolute notation is used if possible, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Force graticule label notation
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.FP1.GRATicule.AXIS.Y.RELative = Var
Var = SCPI.DISPlay.FP1.GRATicule.AXIS.Y.RELative
```

Related Objects

SCPI.DISPlay.FP(Ch).GRATicule.AXIS.Y.STATe

Equivalent Key

FP Menu: **Display** > **Relative Y-Scale**

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:GRATicule:AXIS:Y:RELative {ON|OFF|1|0}

:DISPlay:FP[1-1]:GRATicule:AXIS:Y:RELative?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:FP1:GRATicule:AXIS:Y:RELative 1"
20 OUTPUT 717;":DISPlay:FP1:GRATicule:AXIS:Y:RELative ?"
30 ENTER 717;A
```

SCPI.DISPlay.FP(Ch).GRATicule.AXIS.Y.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.FP(Ch).GRATicule.AXIS.Y.STATe = *Value*

Value = SCPI.DISPlay.FP(Ch).GRATicule.AXIS.Y.STATe

Description

This command sets/gets Y graticule label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Y graticule label
Data Type	Character string type (String)
Range	OFF|SHORt|MIDDle|LONG
Preset Value	SHORt
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.DISPlay.FP1.GRATicule.AXIS.Y.STATe = Var
Var = SCPI.DISPlay.FP1.GRATicule.AXIS.Y.STATe
```

Related Objects

SCPI.DISPlay.FP(Ch).GRATicule.AXIS.Y.RELative

Equivalent Key

FP Menu: **Display** > **Y # of Digits**

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:GRATicule:AXIS:Y:STATe {OFF|SHORt|MIDDle|LONG}

:DISPlay:FP[1-1]:GRATicule:AXIS:Y:STATe?

Query Response

{OFF|SHORt|MIDDle|LONG} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:FP1:GRATicule:AXIS:Y:STATe OFF"
20 OUTPUT 717;":DISPlay:FP1:GRATicule:AXIS:Y:STATe ?"
30 ENTER 717;A$
```

SCPI.DISPlay.FP(Ch).LABel.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.FP(Ch).LABel.DATA = *Value*

Value = SCPI.DISPlay.FP(Ch).LABel.DATA

Description

This command sets/gets Window Title Label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Window Title Label
Data Type	Character string type (String)
Range	254 chars
Preset Value	""
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
SCPI.DISPlay.FP1.LABel.DATA = Var
Var = SCPI.DISPlay.FP1.LABel.DATA
```

Related Objects

SCPI.DISPlay.FP(Ch).LABel.STATe

Equivalent Key

FP Menu: **Display** > **Edit Title Label**

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:LABel:DATA

:DISPlay:FP[1-1]:LABel:DATA?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:FP1:LABel:DATA Sample_Text"
20 OUTPUT 717;":DISPlay:FP1:LABel:DATA ?"
30 ENTER 717;A$
```

SCPI.DISPlay.FP(Ch).LABel.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.FP(Ch).LABel.STATe = *Value*

Value = SCPI.DISPlay.FP(Ch).LABel.STATe

Description

This command show/hides Window Title Label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/hides Window Title Label
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.FP1.LABel.STATe = Var
Var = SCPI.DISPlay.FP1.LABel.STATe
```

Related Objects

SCPI.DISPlay.FP(Ch).LABel.DATA

Equivalent Key

FP Menu: **Display** > **Title Label**

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:LABel:STATe {ON|OFF|1|0}

:DISPlay:FP[1-1]:LABel:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:FP1:LABel:STATe 1"
20 OUTPUT 717;":DISPlay:FP1:LABel:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.FP(Ch).LIMit.FSIGn

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.FP(Ch).LIMit.FSIGn = *Value*

Value = SCPI.DISPlay.FP(Ch).LIMit.FSIGn

Description

This command show/hide fail sign, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	show/hide fail sign
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.FP1.LIMit.FSIGn = Var
Var = SCPI.DISPlay.FP1.LIMit.FSIGn
```

Equivalent Key

FP Menu: **Display** > **Limit Test** > **Fail Sign**

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:LIMit:FSIGn {ON|OFF|1|0}

:DISPlay:FP[1-1]:LIMit:FSIGn?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:FP1:LIMit:FSIGn 1"
20 OUTPUT 717;":DISPlay:FP1:LIMit:FSIGn ?"
30 ENTER 717;A
```

SCPI.DISPlay.FP(Ch).MAXimize

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.FP(Ch).MAXimize = *Value*

Value = SCPI.DISPlay.FP(Ch).MAXimize

Description

This command sets/gets maximize active trace, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Maximize active trace
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.FP1.MAXimize = Var
Var = SCPI.DISPlay.FP1.MAXimize
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:MAXimize {ON|OFF|1|0}

:DISPlay:FP[1-1]:MAXimize?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:FP1:MAXimize 1"
20 OUTPUT 717;":DISPlay:FP1:MAXimize ?"
30 ENTER 717;A
```

SCPI.DISPlay.FP(Ch).SPLit

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.FP(Ch).SPLit = *Value*

Value = SCPI.DISPlay.FP(Ch).SPLit

Description

This command sets/gets split window setting, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Split window setting
Data Type	Character string type (String)
Range	D11_23|D12_34
Preset Value	D11_23
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "D11_23"
SCPI.DISPlay.FP1.SPLit = Var
Var = SCPI.DISPlay.FP1.SPLit
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:SPLit {D11_23|D12_34}

:DISPlay:FP[1-1]:SPLit?

Query Response

{D11_23|D12_34} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:FP1:SPLit D11_23"
20 OUTPUT 717;":DISPlay:FP1:SPLit ?"
30 ENTER 717;A$
```

SCPI.DISPlay.FP(Ch).STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.FP(Ch).STATe = *Value*

Value = SCPI.DISPlay.FP(Ch).STATe

Description

This command sets/gets measurement display on/off. At least one meas window must be turned on., for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Measurement display (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	60, "Failed to hide window" 100, "Option not installed"

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.FP1.STATe = Var
Var = SCPI.DISPlay.FP1.STATe
```

Equivalent Key

PN Menu: **Measurement View** > **Show Window** > **Freq & Power**

SP Menu: **Measurement View** > **Show Window** > **Freq & Power**

FP Menu: **Measurement View** > **Show Window** > **Freq & Power**

TR Menu: **Measurement View** > **Show Window** > **Freq & Power**

AM Menu: **Measurement View** > **Show Window** > **Freq & Power**

BB Menu: **Measurement View** > **Show Window** > **Freq & Power**

USER Menu: **Measurement View** > **Show Window** > **Freq & Power**

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:STATe {ON|OFF|1|0}

:DISPlay:FP[1-1]:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:FP1:STATe 1"
20 OUTPUT 717;":DISPlay:FP1:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.FP(Ch).TABLe.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.FP(Ch).TABLe.STATe = *Value*

Value = SCPI.DISPlay.FP(Ch).TABLe.STATe

Description

This command show/hide text area, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/hide text area
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.FP1.TABLe.STATe = Var
Var = SCPI.DISPlay.FP1.TABLe.STATe
```

Equivalent Key

FP Menu: **Marker** > **Marker List**

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:TABLe[:STATe] {ON|OFF|1|0}

:DISPlay:FP[1-1]:TABLe[:STATe]?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:FP1:TABLe[:STATe] 1"
20 OUTPUT 717;":DISPlay:FP1:TABLe[:STATe] ?"
30 ENTER 717;A
```

SCPI.DISPlay.FP(Ch).TRACe(Tr).LABel.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.FP(Ch).TRACe(Tr).LABel.DATA = *Value*

Value = SCPI.DISPlay.FP(Ch).TRACe(Tr).LABel.DATA

Description

This command sets/gets Trace Title Label, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace Title Label
Data Type	Character string type (String)
Range	254 chars
Preset Value	"Freq"
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
SCPI.DISPlay.FP1.TRACe4.LABel.DATA = Var
Var = SCPI.DISPlay.FP1.TRACe4.LABel.DATA
```

Equivalent Key

FP Menu: **Trace View** > **Trace Label**

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:TRACe[1-4]:LABel:DATA

:DISPlay:FP[1-1]:TRACe[1-4]:LABel:DATA?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:FP1:TRACe4:LABel:DATA Sample_Text"
20 OUTPUT 717;":DISPlay:FP1:TRACe4:LABel:DATA ?"
30 ENTER 717;A$
```

SCPI.DISPlay.FP(Ch).TRACe(Tr).LIMit.LINE

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.FP(Ch).TRACe(Tr).LIMit.LINE = *Value*

Value = SCPI.DISPlay.FP(Ch).TRACe(Tr).LIMit.LINE

Description

This command show/hide limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	show/hide limit line
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.FP1.TRACe4.LIMit.LINE = Var
Var = SCPI.DISPlay.FP1.TRACe4.LIMit.LINE
```

Related Objects

Equivalent Key

FP Menu: **Display** > **Limit Test** > **Limit Line**

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:TRACe[1-4]:LIMit:LINE {ON|OFF|1|0}

:DISPlay:FP[1-1]:TRACe[1-4]:LIMit:LINE?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:FP1:TRACe4:LIMit:LINE 1"
20 OUTPUT 717;":DISPlay:FP1:TRACe4:LIMit:LINE ?"
30 ENTER 717;A
```

SCPI.DISPlay.FP(Ch).TRACe(Tr).MODE

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.FP(Ch).TRACe(Tr).MODE = *Value*

Value = SCPI.DISPlay.FP(Ch).TRACe(Tr).MODE

Description

This command sets/gets show data and/or memory trace, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show data and/or memory trace
Data Type	Character string type (String)
Range	OFF|DATA|MEMory|BOTH
Preset Value	DATA
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.DISPlay.FP1.TRACe4.MODE = Var
Var = SCPI.DISPlay.FP1.TRACe4.MODE
```

Equivalent Key

FP Menu: **Trace View** > **Display Trace**

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:TRACe[1-4]:MODE {OFF|DATA|MEMory|BOTH}

:DISPlay:FP[1-1]:TRACe[1-4]:MODE?

Query Response

{OFF|DATA|MEMory|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:FP1:TRACe4:MODE OFF"
20 OUTPUT 717;":DISPlay:FP1:TRACe4:MODE ?"
30 ENTER 717;A$
```

SCPI.DISPlay.FP(Ch).TRACe(Tr).PERSistence.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.FP(Ch).TRACe(Tr).PERSistence.CLEar

Description

This command clears stored traces, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.DISPlay.FP1.TRACe4.PERSistence.CLEar

Related Objects

SCPI.DISPlay.FP(Ch).TRACe(Tr).PERSistence.STATe

Equivalent Key

FP Menu: **Trace View** > **Persistence** > **Clear Persistent Data**

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:TRACe[1-4]:PERSistence:CLEar

Example of use

10 OUTPUT 717;":DISPlay:FP1:TRACe4:PERSistence:CLEar"

SCPI.DISPlay.FP(Ch).TRACe(Tr).PERSistence.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.FP(Ch).TRACe(Tr).PERSistence.STATe = *Value*

Value = SCPI.DISPlay.FP(Ch).TRACe(Tr).PERSistence.STATe

Description

This command sets/gets trace storage mode, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace storage mode
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.FP1.TRACe4.PERSistence.STATe = Var
Var = SCPI.DISPlay.FP1.TRACe4.PERSistence.STATe
```

Related Objects

SCPI.DISPlay.FP(Ch).TRACe(Tr).PERSistence.CLEar

Equivalent Key

FP Menu: **Trace View** > **Persistence** > **Persistence Mode**

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:TRACe[1-4]:PERSistence:STATe {ON|OFF|1|0}

:DISPlay:FP[1-1]:TRACe[1-4]:PERSistence:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:FP1:TRACe4:PERSistence:STATe 1"
20 OUTPUT 717;":DISPlay:FP1:TRACe4:PERSistence:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.FP(Ch).TRACe(Tr).X.SCALe.AUTO

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.FP(Ch).TRACe(Tr).X.SCALe.AUTO = *Value*

Value = SCPI.DISPlay.FP(Ch).TRACe(Tr).X.SCALe.AUTO

Description

This command sets/gets autoscale setting for X axis, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Autoscale setting for X axis
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.FP1.TRACe4.X.SCALe.AUTO = Var
Var = SCPI.DISPlay.FP1.TRACe4.X.SCALe.AUTO
```

Related Objects

SCPI.DISPlay.FP(Ch).TRACe(Tr).X.SCALe.AUTO

SCPI.DISPlay.FP(Ch).TRACe(Tr).X.SCALe.LEFT

SCPI.DISPlay.FP(Ch).TRACe(Tr).X.SCALe.RIGHt

Equivalent Key

FP Menu: **Scale** > **X Axis** > **Auto**

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:TRACe[1-4]:X[:SCALe]:AUTO {ON|OFF|1|0}

:DISPlay:FP[1-1]:TRACe[1-4]:X[:SCALe]:AUTO?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:FP1:TRACe4:X[:SCALe]:AUTO 1"
20 OUTPUT 717;":DISPlay:FP1:TRACe4:X[:SCALe]:AUTO ?"
30 ENTER 717;A
```

SCPI.DISPlay.FP(Ch).TRACe(Tr).X.SCALe.LEFT

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.FP(Ch).TRACe(Tr).X.SCALe.LEFT = *Value*

Value = SCPI.DISPlay.FP(Ch).TRACe(Tr).X.SCALe.LEFT

Description

This command sets/gets left setting for X axis, for the selected trace *Tr* of the selected channel *Ch*.

NOTE This command is available only when the automatic setting of the X-axis display range (set with SCPI.DISPlay.FP(Ch).TRACe(Tr).X.SCALe.AUTO) is set to OFF.

Variable

Parameter	*Value*
Description	Left setting for X axis
Data Type	Double precision floating point type (Double)
Range	-15 ~ 1.000999999k
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.DISPlay.FP1.TRACe4.X.SCALe.LEFT = Var
Var = SCPI.DISPlay.FP1.TRACe4.X.SCALe.LEFT
```

Related Objects

SCPI.DISPlay.FP(Ch).TRACe(Tr).X.SCALe.AUTO

SCPI.DISPlay.FP(Ch).TRACe(Tr).X.SCALe.RIGHt

Equivalent Key

FP Menu: **Scale** > **X Axis** > **Left**

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:TRACe[1-4]:X[:SCALe]:LEFT {-15 ~ 1.000999999k}

:DISPlay:FP[1-1]:TRACe[1-4]:X[:SCALe]:LEFT?

Query Response

{-15 ~ 1.000999999k} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:FP1:TRACe4:X[:SCALe]:LEFT 0"
20 OUTPUT 717;":DISPlay:FP1:TRACe4:X[:SCALe]:LEFT ?"
30 ENTER 717;A
```

SCPI.DISPlay.FP(Ch).TRACe(Tr).X.SCALe.RIGHt

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.FP(Ch).TRACe(Tr).X.SCALe.RIGHt = *Value*

Value = SCPI.DISPlay.FP(Ch).TRACe(Tr).X.SCALe.RIGHt

Description

This command sets/gets Right setting for X axis, for the selected trace *Tr* of the selected channel *Ch*.

NOTE This command is available only when the automatic setting of the X-axis display range (set with SCPI.DISPlay.FP(Ch).TRACe(Tr).X.SCALe.AUTO) is set to OFF.

Variable

Parameter	*Value*
Description	Right setting for X axis
Data Type	Double precision floating point type (Double)
Range	-14.999999 ~ 1.001k
Preset Value	100u
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.DISPlay.FP1.TRACe4.X.SCALe.RIGHt = Var
Var = SCPI.DISPlay.FP1.TRACe4.X.SCALe.RIGHt
```

Related Objects

SCPI.DISPlay.FP(Ch).TRACe(Tr).X.SCALe.AUTO

SCPI.DISPlay.FP(Ch).TRACe(Tr).X.SCALe.LEFT

Equivalent Key

FP Menu: **Scale** > **X Axis** > **Right**

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:TRACe[1-4]:X[:SCALe]:RIGHt {-14.999999 ~ 1.001k}

:DISPlay:FP[1-1]:TRACe[1-4]:X[:SCALe]:RIGHt?

Query Response

{-14.999999 ~ 1.001k} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:FP1:TRACe4:X[:SCALe]:RIGHt 0"
20 OUTPUT 717;":DISPlay:FP1:TRACe4:X[:SCALe]:RIGHt ?"
30 ENTER 717;A
```

SCPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.AUTO

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.AUTO

Description

This command sets auto scale settings for Y axis, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.DISPlay.FP1.TRACe4.Y.SCALe.AUTO

Related Objects

SCPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.RLEVel

SCPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.RPOSition

Equivalent Key

FP Menu: **Scale** > **Auto Scale**

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:TRACe[1-4]:Y[:SCALe]:AUTO

Example of use

```
10 OUTPUT 717;":DISPlay:FP1:TRACe4:Y[:SCALe]:AUTO"
```

CPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.PDIVision

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.PDIVision = *Value*

Value = SCPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.PDIVision

Description

This command sets/gets scale per division, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale per division
Data Type	Double precision floating point type (Double)
Range	1a ~ 10G
Preset Value	100M
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0.000000000001
SCPI.DISPlay.FP1.TRACe4.Y.SCALe.PDIVision = Var
Var = SCPI.DISPlay.FP1.TRACe4.Y.SCALe.PDIVision
```

Related Objects

SCPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.RLEVel

SCPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.RPOSition

Equivalent Key

FP Menu: **Scale** > **Scale/Div**

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:TRACe[1-4]:Y[:SCALe]:PDIVision {1a ~ 10G}

:DISPlay:FP[1-1]:TRACe[1-4]:Y[:SCALe]:PDIVision?

Query Response

{1a ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:FP1:TRACe4:Y[:SCALe]:PDIVision 0.000000000001"
20 OUTPUT 717;":DISPlay:FP1:TRACe4:Y[:SCALe]:PDIVision ?"
30 ENTER 717;A
```

SCPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.RLEVel

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.RLEVel = *Value*

Value = SCPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.RLEVel

Description

This command sets/gets scale reference level, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale reference level
Data Type	Double precision floating point type (Double)
Range	-500G ~ 500G
Preset Value	1.5G
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -500000000000
SCPI.DISPlay.FP1.TRACe4.Y.SCALe.RLEVel = Var
Var = SCPI.DISPlay.FP1.TRACe4.Y.SCALe.RLEVel
```

Related Objects

SCPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.RPOSition

Equivalent Key

FP Menu: **Scale** > **Reference Value**

FP Menu: **Scale** > **Marker** > **Reference**

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:TRACe[1-4]:Y[:SCALe]:RLEVel {-500G ~ 500G}

:DISPlay:FP[1-1]:TRACe[1-4]:Y[:SCALe]:RLEVel?

Query Response

{-500G ~ 500G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:FP1:TRACe4:Y[:SCALe]:RLEVel -500000000000"
20 OUTPUT 717;":DISPlay:FP1:TRACe4:Y[:SCALe]:RLEVel ?"
30 ENTER 717;A
```

SCPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.RPOSition

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.RPOSition = *Value*

Value = SCPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.RPOSition

Description

This command sets/gets scale reference position, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale reference position
Data Type	Long integer type (Long)
Range	0 ~ 30
Preset Value	5
Unit	Div
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.DISPlay.FP1.TRACe4.Y.SCALe.RPOSition = Var
Var = SCPI.DISPlay.FP1.TRACe4.Y.SCALe.RPOSition
```

Related Objects

SCPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.FP(Ch).TRACe(Tr).Y.SCALe.RLEVel

Equivalent Key

FP Menu: **Scale** > **Reference Position**

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:TRACe[1-4]:Y[:SCALe]:RPOSition {0 ~ 30}

:DISPlay:FP[1-1]:TRACe[1-4]:Y[:SCALe]:RPOSition?

Query Response

{0 ~ 30} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:FP1:TRACe4:Y[:SCALe]:RPOSition 0 "
20 OUTPUT 717;":DISPlay:FP1:TRACe4:Y[:SCALe]:RPOSition ?"
30 ENTER 717;A
```

SCPI.DISPlay.FP(Ch).Y.SCALe.DIVisions

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.FP(Ch).Y.SCALe.DIVisions = *Value*

Value = SCPI.DISPlay.FP(Ch).Y.SCALe.DIVisions

Description

This command sets/gets number of Y division, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Number of Y division
Data Type	Long integer type (Long)
Range	4 ~ 30
Preset Value	10
Unit	-
Resolution	2
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 4
SCPI.DISPlay.FP1.Y.SCALe.DIVisions = Var
Var = SCPI.DISPlay.FP1.Y.SCALe.DIVisions
```

Equivalent Key

FP Menu: **Scale** > **Divisions**

Equivalent SCPI Command

Syntax

:DISPlay:FP[1-1]:Y[:SCALe]:DIVisions {4 ~ 30}

:DISPlay:FP[1-1]:Y[:SCALe]:DIVisions?

Query Response

{4 ~ 30} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:FP1:Y[:SCALe]:DIVisions 4 "
20 OUTPUT 717;":DISPlay:FP1:Y[:SCALe]:DIVisions ?"
30 ENTER 717;A
```

SCPI.DISPlay.IMAGe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.IMAGe = *Value*

Value = SCPI.DISPlay.IMAGe

Description

This command sets/gets image type (normal/inverted).

Variable

Parameter	*Value*
Description	Image type
Data Type	Character string type (String)
Range	NORMal|INVert
Preset Value	NORMal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NORMal"
SCPI.DISPlay.IMAGe = Var
Var = SCPI.DISPlay.IMAGe
```

Equivalent Key

PN Menu: **Display** > **Color Type**

SP Menu: **Display** > **Color Type**

FP Menu: **Display** > **Color Type**

TR Menu: **Display** > **Color Type**

AM Menu: **Display** > **Color Type**

BB Menu: **Display** > **Color Type**

PS Menu: **Display** > **Color Type**

USER Menu: **Display** > **Color Type**

Equivalent SCPI Command

Syntax

:DISPlay:IMAGe {NORMal|INVert}

:DISPlay:IMAGe?

Query Response

{NORMal|INVert} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:IMAGe NORMal"
20 OUTPUT 717;":DISPlay:IMAGe ?"
30 ENTER 717;A$
```

SCPI.DISPlay.LAYout.THRee

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.LAYout.THRee = *Value*

Value = SCPI.DISPlay.LAYout.THRee

Description

This command sets/gets display layout.

Variable

Parameter	*Value*
Description	Display layout
Data Type	Character string type (String)
Range	D12_33|D1_2_3
Preset Value	D12_33
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "D12_33"
SCPI.DISPlay.LAYout.THRee = Var
Var = SCPI.DISPlay.LAYout.THRee
```

Related Objects

SCPI.DISPlay.LAYout.TWO

Equivalent Key

PN Menu: **Measurement View** > **Show Window** > **Layout Window** > **Three Windows**

SP Menu: **Measurement View** > **Show Window** > **Layout Window** > **Three Windows**

FP Menu: **Measurement View** > **Show Window** > **Layout Window** > **Three Windows**

TR Menu: **Measurement View** > **Show Window** > **Layout Window** > **Three Windows**

AM Menu: **Measurement View** > **Show Window** > **Layout Window** > **Three Windows**

BB Menu: **Measurement View** > **Show Window** > **Layout Window** > **Three Windows**

USER Menu: **Measurement View** > **Show Window** > **Layout Window** > **Three Windows**

Equivalent SCPI Command

Syntax

:DISPlay:LAYout:THRee {D12_33|D1_2_3}

:DISPlay:LAYout:THRee?

Query Response

{D12_33|D1_2_3} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:LAYout:THRee D12_33"
20 OUTPUT 717;":DISPlay:LAYout:THRee ?"
30 ENTER 717;A$
```

SCPI.DISPlay.LAYout.TWO

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.LAYout.TWO = *Value*

Value = SCPI.DISPlay.LAYout.TWO

Description

This command sets/gets display layout.

Variable

Parameter	*Value*
Description	Display layout
Data Type	Character string type (String)
Range	D1_2|D12
Preset Value	D1_2
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "D1_2"
SCPI.DISPlay.LAYout.TWO = Var
Var = SCPI.DISPlay.LAYout.TWO
```

Related Objects

SCPI.DISPlay.LAYout.THRee

Equivalent Key

PN Menu: **Measurement View** > **Show Window** > **Layout Window** > **Two Windows**

SP Menu: **Measurement View** > **Show Window** > **Layout Window** > **Two Windows**

FP Menu: **Measurement View** > **Show Window** > **Layout Window** > **Two Windows**

TR Menu: **Measurement View** > **Show Window** > **Layout Window** > **Two Windows**

AM Menu: **Measurement View** > **Show Window** > **Layout Window** > **Two Windows**

BB Menu: **Measurement View** > **Show Window** > **Layout Window** > **Two Windows**

USER Menu: **Measurement View** > **Show Window** > **Layout Window** > **Two Windows**

Equivalent SCPI Command

Syntax

:DISPlay:LAYout:TWO {D1_2|D12}

:DISPlay:LAYout:TWO?

Query Response

{D1_2|D12} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:LAYout:TWO D1_2"
20 OUTPUT 717;":DISPlay:LAYout:TWO ?"
30 ENTER 717;A$
```

SCPI.DISPlay.MAXimize

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.MAXimize = *Value*

Value = SCPI.DISPlay.MAXimize

Description

This command sets/gets maximize value for active instrument window.

Variable

Parameter	*Value*
Description	Active instrument window
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.MAXimize = Var
Var = SCPI.DISPlay.MAXimize
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:MAXimize {ON|OFF|1|0}

:DISPlay:MAXimize?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:MAXimize 1"
20 OUTPUT 717;":DISPlay:MAXimize ?"
30 ENTER 717;A
```

SCPI.DISPlay.MESSage.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.MESSage.CLEar

Description

This command clears caution/message.

Examples

SCPI.DISPlay.MESSage.CLEar

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:MESSage:CLEar

Example of use

10 OUTPUT 717;":DISPlay:MESSage:CLEar"

SCPI.DISPlay.MESSage.DATA[_Q] message_str, as_error

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.MESSage.DATA[_Q] message_str, as_error = *Value*

Value = SCPI.DISPlay.MESSage.DATA[_Q] message_str, as_error

Description

This command sets/gets message window text and color. If second parameter is true, message will be displayed in red color. Messages in white cannot overwrite previous message in red.

Variable

Parameter	*message_str*
Description	Message window string
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*as_error*
Description	Error parameter
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
SCPI.DISPlay.MESSage.DATA[_Q] message_str, as_error = Var
Var = SCPI.DISPlay.MESSage.DATA[_Q] message_str, as_error
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:MESSage:DATA {254 chars ON|OFF|1|0}

:DISPlay:MESSage:DATA?

Query Response

{254 chars ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:MESSage:DATA 1"
20 OUTPUT 717;":DISPlay:MESSage:DATA ?"
```

SCPI.DISPlay.PN(Ch).ALLTrace.PERSistence.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.PN(Ch).ALLTrace.PERSistence.CLEar

Description

This command clears all stored traces, for the selected channel *Ch*.

Examples

SCPI.DISPlay.PN1.ALLTrace.PERSistence.CLEar

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:PN[1-1]:ALLTrace:PERSistence:CLEar

Example of use

10 OUTPUT 717;":DISPlay:PN1:ALLTrace:PERSistence:CLEar"

SCPI.DISPlay.PN(Ch).ANNotation.MARKer.POSition

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PN(Ch).ANNotation.MARKer.POSition = *Value*

Value = SCPI.DISPlay.PN(Ch).ANNotation.MARKer.POSition

Description

This command sets/gets Marker annotation position, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker annotation position
Data Type	Character string type (String)
Range	LEFT|RIGHt
Preset Value	RIGHt
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "LEFT"
SCPI.DISPlay.PN1.ANNotation.MARKer.POSition = Var
Var = SCPI.DISPlay.PN1.ANNotation.MARKer.POSition
```

Related Objects

SCPI.DISPlay.PN(Ch).ANNotation.MEASurement.STATe

Equivalent Key

PN Menu: **Display** > **Marker Information**

Equivalent SCPI Command

Syntax

:DISPlay:PN[1-1]:ANNotation:MARKer:POSition {LEFT|RIGHt}

:DISPlay:PN[1-1]:ANNotation:MARKer:POSition?

Query Response

{LEFT|RIGHt} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PN1:ANNotation:MARKer:POSition LEFT"
20 OUTPUT 717;":DISPlay:PN1:ANNotation:MARKer:POSition ?"
30 ENTER 717;A$
```

SCPI.DISPlay.PN(Ch).ANNotation.MEASurement.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PN(Ch).ANNotation.MEASurement.STATe = *Value*

Value = SCPI.DISPlay.PN(Ch).ANNotation.MEASurement.STATe

Description

This command Show/Hide measurement settings, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/Hide measurement settings
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.PN1.ANNotation.MEASurement.STATe = Var
Var = SCPI.DISPlay.PN1.ANNotation.MEASurement.STATe
```

Related Objects

SCPI.DISPlay.PN(Ch).ANNotation.MARKer.POSition

Equivalent Key

PN Menu: **Display** > **Meas Condition**

Equivalent SCPI Command

Syntax

:DISPlay:PN[1-1]:ANNotation:MEASurement:STATe {ON|OFF|1|0}

:DISPlay:PN[1-1]:ANNotation:MEASurement:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PN1:ANNotation:MEASurement:STATe 1"
20 OUTPUT 717;":DISPlay:PN1:ANNotation:MEASurement:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.PN(Ch).GRATicule.AXIS.Y.RELative

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PN(Ch).GRATicule.AXIS.Y.RELative = *Value*

Value = SCPI.DISPlay.PN(Ch).GRATicule.AXIS.Y.RELative

Description

This command sets/gets force graticule label notation relative. If OFF, absolute notation is used if possible, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Force graticule label notation
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.PN1.GRATicule.AXIS.Y.RELative = Var
Var = SCPI.DISPlay.PN1.GRATicule.AXIS.Y.RELative
```

Related Objects

SCPI.DISPlay.PN(Ch).GRATicule.AXIS.Y.STATe

Equivalent Key

PN Menu: **Display** > **Relative Y-Scale**

Equivalent SCPI Command

Syntax

:DISPlay:PN[1-1]:GRATicule:AXIS:Y:RELative {ON|OFF|1|0}

:DISPlay:PN[1-1]:GRATicule:AXIS:Y:RELative?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PN1:GRATicule:AXIS:Y:RELative 1"
20 OUTPUT 717;":DISPlay:PN1:GRATicule:AXIS:Y:RELative ?"
30 ENTER 717;A
```

SCPI.DISPlay.PN(Ch).GRATicule.AXIS.Y.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PN(Ch).GRATicule.AXIS.Y.STATe = *Value*

Value = SCPI.DISPlay.PN(Ch).GRATicule.AXIS.Y.STATe

Description

This command Show/Hide Y graticule label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/Hide Y graticule label
Data Type	Character string type (String)
Range	OFF|SHORt|MIDDle|LONG
Preset Value	SHORt
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.DISPlay.PN1.GRATicule.AXIS.Y.STATe = Var
Var = SCPI.DISPlay.PN1.GRATicule.AXIS.Y.STATe
```

Related Objects

SCPI.DISPlay.PN(Ch).GRATicule.AXIS.Y.RELative

Equivalent Key

PN Menu: **Display** > **Y # of Digits**

Equivalent SCPI Command

Syntax

:DISPlay:PN[1-1]:GRATicule:AXIS:Y:STATe {OFF|SHORt|MIDDle|LONG}

:DISPlay:PN[1-1]:GRATicule:AXIS:Y:STATe?

Query Response

{OFF|SHORt|MIDDle|LONG} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PN1:GRATicule:AXIS:Y:STATe OFF"
20 OUTPUT 717;":DISPlay:PN1:GRATicule:AXIS:Y:STATe ?"
30 ENTER 717;A$
```

SCPI.DISPlay.PN(Ch).LABel.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PN(Ch).LABel.DATA = *Value*

Value = SCPI.DISPlay.PN(Ch).LABel.DATA

Description

This command sets/gets Window Title Label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Window Title Label
Data Type	Character string type (String)
Range	254 chars
Preset Value	""
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
SCPI.DISPlay.PN1.LABel.DATA = Var
Var = SCPI.DISPlay.PN1.LABel.DATA
```

Related Objects

SCPI.DISPlay.PN(Ch).LABel.STATe

Equivalent Key

PN Menu: **Display** > **Edit Title Label**

Equivalent SCPI Command

Syntax

:DISPlay:PN[1-1]:LABel:DATA

:DISPlay:PN[1-1]:LABel:DATA?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PN1:LABel:DATA Sample_Text"
20 OUTPUT 717;":DISPlay:PN1:LABel:DATA ?"
30 ENTER 717;A$
```

SCPI.DISPlay.PN(Ch).LABel.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PN(Ch).LABel.STATe = *Value*

Value = SCPI.DISPlay.PN(Ch).LABel.STATe

Description

This command Show/Hide Window Title Label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/Hide Window Title Label
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.PN1.LABel.STATe = Var
Var = SCPI.DISPlay.PN1.LABel.STATe
```

Related Objects

Equivalent Key

PN Menu: **Display** > **Title Label**

Equivalent SCPI Command

Syntax

```
:DISPlay:PN[1-1]:LABel:STATe {ON|OFF|1|0}
:DISPlay:PN[1-1]:LABel:STATe?
```

Query Response

```
{ON|OFF|1|0} <newline><^END>
```

Example of use

```
10 OUTPUT 717;":DISPlay:PN1:LABel:STATe 1"
20 OUTPUT 717;":DISPlay:PN1:LABel:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.PN(Ch).LIMit.FSIGn

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PN(Ch).LIMit.FSIGn = *Value*

Value = SCPI.DISPlay.PN(Ch).LIMit.FSIGn

Description

This command show/hide fail sign, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	show/hide fail sign
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.PN1.LIMit.FSIGn = Var
Var = SCPI.DISPlay.PN1.LIMit.FSIGn
```

Equivalent Key

PN Menu: **Display** > **Limit Test** > **Fail Sign**

Equivalent SCPI Command

Syntax

:DISPlay:PN[1-1]:LIMit:FSIGn {ON|OFF|1|0}

:DISPlay:PN[1-1]:LIMit:FSIGn?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PN1:LIMit:FSIGn 1"
20 OUTPUT 717;":DISPlay:PN1:LIMit:FSIGn ?"
30 ENTER 717;A
```

SCPI.DISPlay.PN(Ch).MAXimize

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PN(Ch).MAXimize = *Value*

Value = SCPI.DISPlay.PN(Ch).MAXimize

Description

This command sets/gets maximize active trace, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Maximize active trace
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.PN1.MAXimize = Var
Var = SCPI.DISPlay.PN1.MAXimize
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:PN[1-1]:MAXimize {ON|OFF|1|0}

:DISPlay:PN[1-1]:MAXimize?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PN1:MAXimize 1"
20 OUTPUT 717;":DISPlay:PN1:MAXimize ?"
30 ENTER 717;A
```

SCPI.DISPlay.PN(Ch).STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PN(Ch).STATe = *Value*

Value = SCPI.DISPlay.PN(Ch).STATe

Description

This command sets/gets measurement display on/off. At least one meas window must be turned on., for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Measurement display (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	60, "Failed to hide window" 100, "Option not installed"

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.PN1.STATe = Var
Var = SCPI.DISPlay.PN1.STATe
```

Equivalent Key

PN Menu: **Measurement View** > **Show Window** > **Phase Noise**

SP Menu: **Measurement View** > **Show Window** > **Phase Noise**

FP Menu: **Measurement View** > **Show Window** > **Phase Noise**

TR Menu: **Measurement View** > **Show Window** > **Phase Noise**

AM Menu: **Measurement View** > **Show Window** > **Phase Noise**

BB Menu: **Measurement View** > **Show Window** > **Phase Noise**

USER Menu: **Measurement View** > **Show Window** > **Phase Noise**

Equivalent SCPI Command

Syntax

:DISPlay:PN[1-1]:STATe {ON|OFF|1|0}

:DISPlay:PN[1-1]:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PN1:STATe 1"
20 OUTPUT 717;":DISPlay:PN1:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.PN(Ch).TABLe.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PN(Ch).TABLe.STATe = *Value*

Value = SCPI.DISPlay.PN(Ch).TABLe.STATe

Description

This command show/hide text area, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	show/hide text area
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.PN1.TABLe.STATe = Var
Var = SCPI.DISPlay.PN1.TABLe.STATe
```

Equivalent Key

PN Menu: **Marker** > **Marker List**

Equivalent SCPI Command

Syntax

:DISPlay:PN[1-1]:TABLe[:STATe] {ON|OFF|1|0}

:DISPlay:PN[1-1]:TABLe[:STATe]?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PN1:TABLe[:STATe] 1"
20 OUTPUT 717;":DISPlay:PN1:TABLe[:STATe] ?"
30 ENTER 717;A
```

SCPI.DISPlay.PN(Ch).TRACe(Tr).LABel.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PN(Ch).TRACe(Tr).LABel.DATA = *Value*

Value = SCPI.DISPlay.PN(Ch).TRACe(Tr).LABel.DATA

Description

This command sets/gets Trace Title Label, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace Title Label
Data Type	Character string type (String)
Range	254 chars
Preset Value	"Phase Noise"
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
SCPI.DISPlay.PN1.TRACe1.LABel.DATA = Var
Var = SCPI.DISPlay.PN1.TRACe1.LABel.DATA
```

Equivalent Key

PN Menu: **Trace View** > **Trace Label**

Equivalent SCPI Command

Syntax

:DISPlay:PN[1-1]:TRACe[1-1]:LABel:DATA

:DISPlay:PN[1-1]:TRACe[1-1]:LABel:DATA?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PN1:TRACe1:LABel:DATA Sample_Text"
20 OUTPUT 717;":DISPlay:PN1:TRACe1:LABel:DATA ?"
30 ENTER 717;A$
```

SCPI.DISPlay.PN(Ch).TRACe(Tr).LIMit.LINE

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PN(Ch).TRACe(Tr).LIMit.LINE = *Value*

Value = SCPI.DISPlay.PN(Ch).TRACe(Tr).LIMit.LINE

Description

This command sets/gets show/hide limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/hide limit line
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.PN1.TRACe1.LIMit.LINE = Var
Var = SCPI.DISPlay.PN1.TRACe1.LIMit.LINE
```

Equivalent Key

PN Menu: **Display** > **Limit Test** > **Limit Line**

Equivalent SCPI Command

Syntax

:DISPlay:PN[1-1]:TRACe[1-1]:LIMit:LINE {ON|OFF|1|0}

:DISPlay:PN[1-1]:TRACe[1-1]:LIMit:LINE?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PN1:TRACe1:LIMit:LINE 1"
20 OUTPUT 717;":DISPlay:PN1:TRACe1:LIMit:LINE ?"
30 ENTER 717;A
```

SCPI.DISPlay.PN(Ch).TRACe(Tr).MODE

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PN(Ch).TRACe(Tr).MODE = *Value*

Value = SCPI.DISPlay.PN(Ch).TRACe(Tr).MODE

Description

This command sets/gets data and/or memory trace, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Data and/or memory trace
Data Type	Character string type (String)
Range	OFF|DATA|MEMory|BOTH
Preset Value	DATA
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.DISPlay.PN1.TRACe1.MODE = Var
Var = SCPI.DISPlay.PN1.TRACe1.MODE
```

Equivalent Key

PN Menu: **Trace View** > **Display Trace**

Equivalent SCPI Command

Syntax

:DISPlay:PN[1-1]:TRACe[1-1]:MODE {OFF|DATA|MEMory|BOTH}

:DISPlay:PN[1-1]:TRACe[1-1]:MODE?

Query Response

{OFF|DATA|MEMory|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PN1:TRACe1:MODE OFF"
20 OUTPUT 717;":DISPlay:PN1:TRACe1:MODE ?"
30 ENTER 717;A$
```

SCPI.DISPlay.PN(Ch).TRACe(Tr).PERSistence.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.PN(Ch).TRACe(Tr).PERSistence.CLEar

Description

This command clears stored traces, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.DISPlay.PN1.TRACe1.PERSistence.CLEar

Related Objects

SCPI.DISPlay.PN(Ch).TRACe(Tr).PERSistence.STATe

Equivalent Key

PN Menu: **Trace View** > **Persistence** > **Clear Persistent Data**

Equivalent SCPI Command

Syntax

:DISPlay:PN[1-1]:TRACe[1-1]:PERSistence:CLEar

Example of use

```
10 OUTPUT 717;":DISPlay:PN1:TRACe1:PERSistence:CLEar"
```

SCPI.DISPlay.PN(Ch).TRACe(Tr).PERSistence.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PN(Ch).TRACe(Tr).PERSistence.STATe = *Value*

Value = SCPI.DISPlay.PN(Ch).TRACe(Tr).PERSistence.STATe

Description

This command sets/gets trace storage mode, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace storage mode
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.PN1.TRACe1.PERSistence.STATe = Var
Var = SCPI.DISPlay.PN1.TRACe1.PERSistence.STATe
```

Related Objects

SCPI.DISPlay.PN(Ch).TRACe(Tr).PERSistence.CLEar

Equivalent Key

PN Menu: **Trace View** > **Persistence** > **Persistence Mode**

Equivalent SCPI Command

Syntax

:DISPlay:PN[1-1]:TRACe[1-1]:PERSistence:STATe {ON|OFF|1|0}

:DISPlay:PN[1-1]:TRACe[1-1]:PERSistence:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PN1:TRACe1:PERSistence:STATe 1"
20 OUTPUT 717;":DISPlay:PN1:TRACe1:PERSistence:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.PN(Ch).TRACe(Tr).X.SCALe.AUTO

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PN(Ch).TRACe(Tr).X.SCALe.AUTO = *Value*

Value = SCPI.DISPlay.PN(Ch).TRACe(Tr).X.SCALe.AUTO

Description

This command sets/gets autoscale setting for X axis, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Autoscale setting for X axis
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.PN1.TRACe1.X.SCALe.AUTO = Var
Var = SCPI.DISPlay.PN1.TRACe1.X.SCALe.AUTO
```

Related Objects

SCPI.DISPlay.PN(Ch).TRACe(Tr).X.SCALe.LEFT

SCPI.DISPlay.PN(Ch).TRACe(Tr).X.SCALe.RIGHt

Equivalent Key

PN Menu: **Scale** > **X Axis** > **Auto**

Equivalent SCPI Command

Syntax

:DISPlay:PN[1-1]:TRACe[1-1]:X[:SCALe]:AUTO {ON|OFF|1|0}

:DISPlay:PN[1-1]:TRACe[1-1]:X[:SCALe]:AUTO?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PN1:TRACe1:X[:SCALe]:AUTO 1"
20 OUTPUT 717;":DISPlay:PN1:TRACe1:X[:SCALe]:AUTO ?"
30 ENTER 717;A
```

SCPI.DISPlay.PN(Ch).TRACe(Tr).X.SCALe.LEFT

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PN(Ch).TRACe(Tr).X.SCALe.LEFT = *Value*

Value = SCPI.DISPlay.PN(Ch).TRACe(Tr).X.SCALe.LEFT

Description

This command sets/gets left setting for X axis, for the selected trace *Tr* of the selected channel *Ch*.

NOTE This command is available only when the automatic setting of the X-axis display range (set with SCPI.DISPlay.PN(Ch).TRACe(Tr).X.SCALe.AUTO is set to OFF.

Variable

Parameter	*Value*
Description	Left setting for X axis
Data Type	Double precision floating point type (Double)
Range	1 ~ 99.9999999M
Preset Value	1k
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.DISPlay.PN1.TRACe1.X.SCALe.LEFT = Var
Var = SCPI.DISPlay.PN1.TRACe1.X.SCALe.LEFT
```

Related Objects

SCPI.DISPlay.PN(Ch).TRACe(Tr).X.SCALe.AUTO

SCPI.DISPlay.PN(Ch).TRACe(Tr).X.SCALe.RIGHt

Equivalent Key

PN Menu: **Scale** > **X Axis** > **Left**

Equivalent SCPI Command

Syntax

:DISPlay:PN[1-1]:TRACe[1-1]:X[:SCALe]:LEFT {1 ~ 99.9999999M}

:DISPlay:PN[1-1]:TRACe[1-1]:X[:SCALe]:LEFT?

Query Response

{1 ~ 99.9999999M} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PN1:TRACe1:X[:SCALe]:LEFT 0"
20 OUTPUT 717;":DISPlay:PN1:TRACe1:X[:SCALe]:LEFT ?"
30 ENTER 717;A
```

SCPI.DISPlay.PN(Ch).TRACe(Tr).X.SCALe.RIGHt

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PN(Ch).TRACe(Tr).X.SCALe.RIGHt = *Value*

Value = SCPI.DISPlay.PN(Ch).TRACe(Tr).X.SCALe.RIGHt

Description

This command sets/gets right setting for X axis, for the selected trace *Tr* of the selected channel *Ch*.

NOTE This command is available only when the automatic setting of the X-axis display range (set with SCPI.DISPlay.PN(Ch).TRACe(Tr).X.SCALe.AUTO is set to OFF.

Variable

Parameter	*Value*
Description	Right setting for X axis
Data Type	Double precision floating point type (Double)
Range	1.1 ~ 100M
Preset Value	10M
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.DISPlay.PN1.TRACe1.X.SCALe.RIGHt = Var
Var = SCPI.DISPlay.PN1.TRACe1.X.SCALe.RIGHt
```

Related Objects

SCPI.DISPlay.PN(Ch).TRACe(Tr).X.SCALe.AUTO

SCPI.DISPlay.PN(Ch).TRACe(Tr).X.SCALe.LEFT

Equivalent Key

PN Menu: **Scale** > **X Axis** > **Right**

Equivalent SCPI Command

Syntax

:DISPlay:PN[1-1]:TRACe[1-1]:X[:SCALe]:RIGHt {1.1 ~ 100M}

:DISPlay:PN[1-1]:TRACe[1-1]:X[:SCALe]:RIGHt?

Query Response

{1.1 ~ 100M} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PN1:TRACe1:X[:SCALe]:RIGHt 0"
20 OUTPUT 717;":DISPlay:PN1:TRACe1:X[:SCALe]:RIGHt ?"
30 ENTER 717;A
```

SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.AUTO

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.AUTO

Description

This command sets auto scale setting for Y axis, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.DISPlay.PN1.TRACe1.Y.SCALe.AUTO

Related Objects

SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.RLEVel

SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.RPOSition

Equivalent Key

PN Menu: **Scale** > **Auto Scale**

Equivalent SCPI Command

Syntax

:DISPlay:PN[1-1]:TRACe[1-1]:Y[:SCALe]:AUTO

Example of use

```
10 OUTPUT 717;":DISPlay:PN1:TRACe1:Y[:SCALe]:AUTO"
```

SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.PDIVision

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.PDIVision = *Value*

Value = SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.PDIVision

Description

This command sets/gets scale per division, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale per division
Data Type	Double precision floating point type (Double)
Range	1a ~ 10G
Preset Value	10
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0.000000000001
SCPI.DISPlay.PN1.TRACe1.Y.SCALe.PDIVision = Var
Var = SCPI.DISPlay.PN1.TRACe1.Y.SCALe.PDIVision
```

Related Objects

SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.RLEVel

SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.RPOSition

Equivalent Key

PN Menu: **Scale** > **Scale/Div**

Equivalent SCPI Command

Syntax

:DISPlay:PN[1-1]:TRACe[1-1]:Y[:SCALe]:PDIVision {1a ~ 10G}

:DISPlay:PN[1-1]:TRACe[1-1]:Y[:SCALe]:PDIVision?

Query Response

{1a ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PN1:TRACe1:Y[:SCALe]:PDIVision 0.000000000001"
20 OUTPUT 717;":DISPlay:PN1:TRACe1:Y[:SCALe]:PDIVision ?"
30 ENTER 717;A
```

SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.RLEVel

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.RLEVel = *Value*

Value = SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.RLEVel

Description

This command sets/gets scale reference level, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale reference level
Data Type	Double precision floating point type (Double)
Range	-50G ~ 50G
Preset Value	-20
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -50000000000
SCPI.DISPlay.PN1.TRACe1.Y.SCALe.RLEVel = Var
Var = SCPI.DISPlay.PN1.TRACe1.Y.SCALe.RLEVel
```

Related Objects

SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.RPOSition

Equivalent Key

PN Menu: **Scale** > **Reference Value**

PN Menu: **Scale** > **Marker** > **Reference**

Equivalent SCPI Command

Syntax

:DISPlay:PN[1-1]:TRACe[1-1]:Y[:SCALe]:RLEVel {-50G ~ 50G}

:DISPlay:PN[1-1]:TRACe[1-1]:Y[:SCALe]:RLEVel?

Query Response

{-50G ~ 50G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PN1:TRACe1:Y[:SCALe]:RLEVel -50000000000"
20 OUTPUT 717;":DISPlay:PN1:TRACe1:Y[:SCALe]:RLEVel ?"
30 ENTER 717;A
```

SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.RPOSition

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.RPOSition = *Value*

Value = SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.RPOSition

Description

This command sets/gets scale reference position, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale reference position
Data Type	Long integer type (Long)
Range	0 ~ 30
Preset Value	16
Unit	Div
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.DISPlay.PN1.TRACe1.Y.SCALe.RPOSition = Var
Var = SCPI.DISPlay.PN1.TRACe1.Y.SCALe.RPOSition
```

Related Objects

SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.PN(Ch).TRACe(Tr).Y.SCALe.RLEVel

Equivalent Key

PN Menu: **Scale** > **Reference Position**

Equivalent SCPI Command

Syntax

:DISPlay:PN[1-1]:TRACe[1-1]:Y[:SCALe]:RPOSition {0 ~ 30}

:DISPlay:PN[1-1]:TRACe[1-1]:Y[:SCALe]:RPOSition?

Query Response

{0 ~ 30} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PN1:TRACe1:Y[:SCALe]:RPOSition 0 "
20 OUTPUT 717;":DISPlay:PN1:TRACe1:Y[:SCALe]:RPOSition ?"
30 ENTER 717;A
```

SCPI.DISPlay.PN(Ch).Y.SCALe.DIVisions

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PN(Ch).Y.SCALe.DIVisions = *Value*

Value = SCPI.DISPlay.PN(Ch).Y.SCALe.DIVisions

Description

This command sets/gets number of Y division, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Number of Y division
Data Type	Long integer type (Long)
Range	4 ~ 30
Preset Value	16
Unit	-
Resolution	2
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 4
SCPI.DISPlay.PN1.Y.SCALe.DIVisions = Var
Var = SCPI.DISPlay.PN1.Y.SCALe.DIVisions
```

Equivalent Key

PN Menu: **Scale** > **Divisions**

Equivalent SCPI Command

Syntax

```
:DISPlay:PN[1-1]:Y[:SCALe]:DIVisions {4 ~ 30}
:DISPlay:PN[1-1]:Y[:SCALe]:DIVisions?
```

Query Response

```
{4 ~ 30} <newline><^END>
```

Example of use

```
10 OUTPUT 717;":DISPlay:PN1:Y[:SCALe]:DIVisions 4 "
20 OUTPUT 717;":DISPlay:PN1:Y[:SCALe]:DIVisions ?"
30 ENTER 717;A
```

SCPI.DISPlay.PROGress.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PROGress.STATe = *Value*

Value = SCPI.DISPlay.PROGress.STATe

Description

This command sets/gets progress display update.

Variable

Parameter	*Value*
Description	Progress display update
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.PROGress.STATe = Var
Var = SCPI.DISPlay.PROGress.STATe
```

Equivalent Key

PN Menu: **Display** > **Progress Indicator**

SP Menu: **Display** > **Progress Indicator**

FP Menu: **Display** > **Progress Indicator**

TR Menu: **Display** > **Progress Indicator**

AM Menu: **Display** > **Progress Indicator**

BB Menu: **Display** > **Progress Indicator**

PS Menu: **Display** > **Progress Indicator**

USER Menu: **Display** > **Progress Indicator**

Equivalent SCPI Command

Syntax

:DISPlay:PROGress:STATe {ON|OFF|1|0}

:DISPlay:PROGress:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PROGress:STATe 1"
20 OUTPUT 717;":DISPlay:PROGress:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.PS(Ch).ALLTrace.PERSistence.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.PS(Ch).ALLTrace.PERSistence.CLEar

Description

This command clears all stored traces, for the selected channel *Ch*.

Examples

SCPI.DISPlay.PS1.ALLTrace.PERSistence.CLEar

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:PS[1-1]:ALLTrace:PERSistence:CLEar

Example of use

10 OUTPUT 717;":DISPlay:PS1:ALLTrace:PERSistence:CLEar"

SCPI.DISPlay.PS(Ch).ANNotation.MARKer.POSition

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PS(Ch).ANNotation.MARKer.POSition = *Value*

Value = SCPI.DISPlay.PS(Ch).ANNotation.MARKer.POSition

Description

This command sets/gets Marker annotation position, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker annotation position
Data Type	Character string type (String)
Range	LEFT|RIGHt
Preset Value	RIGHt
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "LEFT"
SCPI.DISPlay.PS1.ANNotation.MARKer.POSition = Var
Var = SCPI.DISPlay.PS1.ANNotation.MARKer.POSition
```

Related Objects

SCPI.DISPlay.PS(Ch).ANNotation.MEASurement.STATe

Equivalent Key

PS Menu: **Display** > **Marker Information**

Equivalent SCPI Command

Syntax

:DISPlay:PS[1-1]:ANNotation:MARKer:POSition {LEFT|RIGHt}

:DISPlay:PS[1-1]:ANNotation:MARKer:POSition?

Query Response

{LEFT|RIGHt} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PS1:ANNotation:MARKer:POSition LEFT"
20 OUTPUT 717;":DISPlay:PS1:ANNotation:MARKer:POSition?"
30 ENTER 717;A$
```

SCPI.DISPlay.PS(Ch).ANNotation.MEASurement.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PS(Ch).ANNotation.MEASurement.STATe = *Value*

Value = SCPI.DISPlay.PS(Ch).ANNotation.MEASurement.STATe

Description

This command Show/Hide measurement settings, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/Hide measurement settings
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.PS1.ANNotation.MEASurement.STATe = Var
Var = SCPI.DISPlay.PS1.ANNotation.MEASurement.STATe
```

Related Objects

SCPI.DISPlay.PS(Ch).ANNotation.MARKer.POSition

Equivalent Key

PS Menu: **Display** > **Meas Condition**

Equivalent SCPI Command

Syntax

:DISPlay:PS[1-1]:ANNotation:MEASurement:STATe {ON|OFF|1|0}

:DISPlay:PS[1-1]:ANNotation:MEASurement:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PS1:ANNotation:MEASurement:STATe 1"
20 OUTPUT 717;":DISPlay:PS1:ANNotation:MEASurement:STATe?"
30 ENTER 717;A
```

SCPI.DISPlay.PS(Ch).GRATicule.AXIS.Y.RELative

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PS(Ch).GRATicule.AXIS.Y.RELative = *Value*

Value = SCPI.DISPlay.PS(Ch).GRATicule.AXIS.Y.RELative

Description

This command sets/gets force graticule label notation relative. If OFF, absolute notation is used if possible, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Force graticule label notation
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.PS1.GRATicule.AXIS.Y.RELative = Var
Var = SCPI.DISPlay.PS1.GRATicule.AXIS.Y.RELative
```

Related Objects

SCPI.DISPlay.PS(Ch).GRATicule.AXIS.Y.STATe

Equivalent Key

PS Menu: **Display** > **Relative Y-Scale**

Equivalent SCPI Command

Syntax

:DISPlay:PS[1-1]:GRATicule:AXIS:Y:RELative {ON|OFF|1|0}

:DISPlay:PS[1-1]:GRATicule:AXIS:Y:RELative?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PS1:GRATicule:AXIS:Y:RELative 1"
20 OUTPUT 717;":DISPlay:PS1:GRATicule:AXIS:Y:RELative?"
30 ENTER 717;A
```

SCPI.DISPlay.PS(Ch).GRATicule.AXIS.Y.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PS(Ch).GRATicule.AXIS.Y.STATe = *Value*

Value = SCPI.DISPlay.PS(Ch).GRATicule.AXIS.Y.STATe

Description

This command Show/Hide Y graticule label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/Hide Y graticule label
Data Type	Character string type (String)
Range	OFF|SHORt|MIDDle|LONG
Preset Value	SHORt
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.DISPlay.PS1.GRATicule.AXIS.Y.STATe = Var
Var = SCPI.DISPlay.PS1.GRATicule.AXIS.Y.STATe
```

Related Objects

SCPI.DISPlay.PS(Ch).GRATicule.AXIS.Y.RELative

Equivalent Key

PS Menu: **Display** > **Y # of Digits**

Equivalent SCPI Command

Syntax

:DISPlay:PS[1-1]:GRATicule:AXIS:Y:STATe {OFF|SHORt|MIDDle|LONG}

:DISPlay:PS[1-1]:GRATicule:AXIS:Y:STATe?

Query Response

{OFF|SHORt|MIDDle|LONG} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PS1:GRATicule:AXIS:Y:STATe OFF"
20 OUTPUT 717;":DISPlay:PS1:GRATicule:AXIS:Y:STATe?"
30 ENTER 717;A$
```

SCPI.DISPlay.PS(Ch).LABel.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PS(Ch).LABel.DATA = *Value*

Value = SCPI.DISPlay.PS(Ch).LABel.DATA

Description

This command sets/gets Window Title Label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Window Title Label
Data Type	Character string type (String)
Range	254 chars
Preset Value	""
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var = "Sample_Text"
SCPI.DISPlay.PS1.LABel.DATA = Var
Var = SCPI.DISPlay.PS1.LABel.DATA
```

Related Objects

SCPI.DISPlay.PS(Ch).LABel.STATe

Equivalent Key

PS Menu: **Display** > **Edit Title Label**

Equivalent SCPI Command

Syntax

:DISPlay:PS[1-1]:LABel:DATA

:DISPlay:PS[1-1]:LABel:DATA?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PS1:LABel:DATA Sample_Text"
20 OUTPUT 717;":DISPlay:PS1:LABel:DATA?"
30 ENTER 717;A$
```

SCPI.DISPlay.PS(Ch).LABel.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PS(Ch).LABel.STATe = *Value*

Value = SCPI.DISPlay.PS(Ch).LABel.STATe

Description

This command Show/Hide Window Title Label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/Hide Window Title Label
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.PS1.LABel.STATe = Var
Var = SCPI.DISPlay.PS1.LABel.STATe
```

Related Objects

SCPI.DISPlay.PS(Ch).LABel.DATA

Equivalent Key

PS Menu: **Display** > **Title Label**

Equivalent SCPI Command

Syntax

:DISPlay:PS[1-1]:LABel:STATe {ON|OFF|1|0}

:DISPlay:PS[1-1]:LABel:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PS1:LABel:STATe 1"
20 OUTPUT 717;":DISPlay:PS1:LABel:STATe?"
30 ENTER 717;A
```

SCPI.DISPlay.PS(Ch).LIMit.FSIGn

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PS(Ch).LIMit.FSIGn = *Value*

Value = SCPI.DISPlay.PS(Ch).LIMit.FSIGn

Description

This command show/hide fail sign, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	show/hide fail sign
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.PS1.LIMit.FSIGn = Var
Var = SCPI.DISPlay.PS1.LIMit.FSIGn
```

Equivalent Key

PS Menu: **Display** > **Limit Test** > **Fail Sign**

Equivalent SCPI Command

Syntax

```
:DISPlay:PS[1-1]:LIMit:FSIGn {ON|OFF|1|0}
:DISPlay:PS[1-1]:LIMit:FSIGn?
```

Query Response

```
{ON|OFF|1|0} <newline><^END>
```

Example of use

```
10 OUTPUT 717;":DISPlay:PS1:LIMit:FSIGn 1"
20 OUTPUT 717;":DISPlay:PS1:LIMit:FSIGn?"
30 ENTER 717;A
```

SCPI.DISPlay.PS(Ch).MAXimize

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PS(Ch).MAXimize = *Value*

Value = SCPI.DISPlay.PS(Ch).MAXimize

Description

This command sets/gets maximize active trace, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Maximize active trace
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.PS1.MAXimize = Var
Var = SCPI.DISPlay.PS1.MAXimize
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:PS[1-1]:MAXimize {ON|OFF|1|0}

:DISPlay:PS[1-1]:MAXimize?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PS1:MAXimize 1"
20 OUTPUT 717;":DISPlay:PS1:MAXimize?"
30 ENTER 717;A
```

SCPI.DISPlay.PS(Ch).STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PS(Ch).STATe = *Value*

Value = SCPI.DISPlay.PS(Ch).STATe

Description

This command sets/gets measurement display on/off. At least one meas window must be turned on., for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Measurement display (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	60, "Failed to hide window" 100, "Option not installed"

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.PS1.STATe = Var
Var = SCPI.DISPlay.PS1.STATe
```

Equivalent Key

PN Menu: **Measurement View** > **Show Window** > **Segment PN**

SP Menu: **Measurement View** > **Show Window** > **Segment PN**

FP Menu: **Measurement View** > **Show Window** > **Segment PN**

TR Menu: **Measurement View** > **Show Window** > **Segment PN**

AM Menu: **Measurement View** > **Show Window** > **Segment PN**

PS Menu: **Measurement View** > **Show Window** > **Segment PN**

BB Menu: **Measurement View** > **Show Window** > **Segment PN**

USER Menu: **Measurement View** > **Show Window** > **Segment PN**

Equivalent SCPI Command

Syntax

:DISPlay:PS[1-1]:STATe {ON|OFF|1|0}

:DISPlay:PS[1-1]:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PS1:STATe 1"
20 OUTPUT 717;":DISPlay:PS1:STATe?"
30 ENTER 717;A
```

SCPI.DISPlay.PS(Ch).TABLe.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PS(Ch).TABLe.STATe = *Value*

Value = SCPI.DISPlay.PS(Ch).TABLe.STATe

Description

This command show/hide text area, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	show/hide text area
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.PS1.TABLe.STATe = Var
Var = SCPI.DISPlay.PS1.TABLe.STATe
```

Equivalent Key

PS Menu: **Marker** > **Marker List**

Equivalent SCPI Command

Syntax

:DISPlay:PS[1-1]:TABLe[:STATe] {ON|OFF|1|0}

:DISPlay:PS[1-1]:TABLe[:STATe]?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PS1:TABLe[:STATe] 1"
20 OUTPUT 717;":DISPlay:PS1:TABLe[:STATe]?"
30 ENTER 717;A
```

SCPI.DISPlay.PS(Ch).TRACe(Tr).LABel.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PS(Ch).TRACe(Tr).LABel.DATA = *Value*

Value = SCPI.DISPlay.PS(Ch).TRACe(Tr).LABel.DATA

Description

This command sets/gets Trace Title Label, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace Title Label
Data Type	Character string type (String)
Range	254 chars
Preset Value	"Segment PN"
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var = "Sample_Test"
SCPI.DISPlay.PS1.TRACe1.LABel.DATA = Var
Var = SCPI.DISPlay.PS1.TRACe1.LABel.DATA
```

Equivalent Key

PS Menu: **Trace View** > **Trace Label**

Equivalent SCPI Command

Syntax

```
:DISPlay:PS[1-1]:TRACe[1-1]:LABel:DATA
:DISPlay:PS[1-1]:TRACe[1-1]:LABel:DATA?
```

Query Response

```
{String}<newline><^END>
```

Example of use

```
10 OUTPUT 717;":DISPlay:PS1:TRACe1:LABel:DATA Sample_Text"
20 OUTPUT 717;":DISPlay:PS1:TRACe1:LABel:DATA?"
30 ENTER 717;A$
```

SCPI.DISPlay.PS(Ch).TRACe(Tr).LIMit.LINE

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PS(Ch).TRACe(Tr).LIMit.LINE = *Value*

Value = SCPI.DISPlay.PS(Ch).TRACe(Tr).LIMit.LINE

Description

This command sets/gets show/hide limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/hide limit line
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.PS1.TRACe1.LIMit.LINE = Var
Var = SCPI.DISPlay.PS1.TRACe1.LIMit.LINE
```

Equivalent Key

PS Menu: **Display** > **Limit Test** > **Limit Line**

Equivalent SCPI Command

Syntax

:DISPlay:PS[1-1]:TRACe[1-1]:LIMit:LINE {ON|OFF|1|0}

:DISPlay:PS[1-1]:TRACe[1-1]:LIMit:LINE?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PS1:TRACe1:LIMit:LINE 1"
20 OUTPUT 717;":DISPlay:PS1:TRACe1:LIMit:LINE?"
30 ENTER 717;A
```

SCPI.DISPlay.PS(Ch).TRACe(Tr).MODE

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PS(Ch).TRACe(Tr).MODE = *Value*

Value = SCPI.DISPlay.PS(Ch).TRACe(Tr).MODE

Description

This command sets/gets data and/or memory trace, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Data and/or memory trace
Data Type	Character string type (String)
Range	OFF|DATA|MEMory|BOTH
Preset Value	DATA
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.DISPlay.PS1.TRACe1.MODE = Var
Var = SCPI.DISPlay.PS1.TRACe1.MODE
```

Equivalent Key

PS Menu: **Trace View** > **Display Trace**

Equivalent SCPI Command

Syntax

:DISPlay:PS[1-1]:TRACe[1-1]:MODE {OFF|DATA|MEMory|BOTH}

:DISPlay:PS[1-1]:TRACe[1-1]:MODE?

Query Response

{OFF|DATA|MEMory|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PS1:TRACe1:MODE OFF"
20 OUTPUT 717;":DISPlay:PS1:TRACe1:MODE ?"
30 ENTER 717;A$
```

SCPI.DISPlay.PS(Ch).TRACe(Tr).PERSistence.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.PS(Ch).TRACe(Tr).PERSistence.CLEar

Description

This command clears stored traces, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.DISPlay.PS1.TRACe1.PERSistence.CLEar

Related Objects

SCPI.DISPlay.PS(Ch).TRACe(Tr).PERSistence.STATe

Equivalent Key

PS Menu: **Trace View** > **Persistence** > **Clear Persistent Data**

Equivalent SCPI Command

Syntax

:DISPlay:PS[1-1]:TRACe[1-1]:PERSistence:CLEar

Example of use

```
10 OUTPUT 717;":DISPlay:PS1:TRACe1:PERSistence:CLEar"
```

SCPI.DISPlay.PS(Ch).TRACe(Tr).PERSistence.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PS(Ch).TRACe(Tr).PERSistence.STATe = *Value*

Value = SCPI.DISPlay.PS(Ch).TRACe(Tr).PERSistence.STATe

Description

This command sets/gets trace storage mode, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace storage mode
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.PS1.TRACe1.PERSistence.STATe = Var
Var = SCPI.DISPlay.PS1.TRACe1.PERSistence.STATe
```

Related Objects

SCPI.DISPlay.PS(Ch).TRACe(Tr).PERSistence.CLEar

Equivalent Key

PS Menu: **Trace View** > **Persistence** > **Persistence Mode**

Equivalent SCPI Command

Syntax

:DISPlay:PS[1-1]:TRACe[1-1]:PERSistence:STATe {ON|OFF|1|0}

:DISPlay:PS[1-1]:TRACe[1-1]:PERSistence:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PS1:TRACe1:PERSistence:STATe 1"
20 OUTPUT 717;":DISPlay:PS1:TRACe1:PERSistence:STATe?"
30 ENTER 717;A
```

SCPI.DISPlay.PS(Ch).TRACe(Tr).X.SCALe.AUTO

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PS(Ch).TRACe(Tr).X.SCALe.AUTO = *Value*

Value = SCPI.DISPlay.PS(Ch).TRACe(Tr).X.SCALe.AUTO

Description

This command sets/gets autoscale setting for X axis, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Autoscale setting for X axis
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.PS1.TRACe1.X.SCALe.AUTO = Var
Var = SCPI.DISPlay.PS1.TRACe1.X.SCALe.AUTO
```

Related Objects

SCPI.DISPlay.PS(Ch).TRACe(Tr).X.SCALe.LEFT

SCPI.DISPlay.PS(Ch).TRACe(Tr).X.SCALe.RIGHt

Equivalent Key

PS Menu: **Scale** > **X Axis** > **Auto**

Equivalent SCPI Command

Syntax

:DISPlay:PS[1-1]:TRACe[1-1]:X[:SCALe]:AUTO {ON|OFF|1|0}

:DISPlay:PS[1-1]:TRACe[1-1]:X[:SCALe]:AUTO?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PS1:TRACe1:X[:SCALe]:AUTO 1"
20 OUTPUT 717;":DISPlay:PS1:TRACe1:X[:SCALe]:AUTO?"
30 ENTER 717;A
```

SCPI.DISPlay.PS(Ch).TRACe(Tr).X.SCALe.LEFT

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PS(Ch).TRACe(Tr).X.SCALe.LEFT = *Value*

Value = SCPI.DISPlay.PS(Ch).TRACe(Tr).X.SCALe.LEFT

Description

This command sets/gets left setting for X axis, for the selected trace *Tr* of the selected channel *Ch*.

NOTE This command is available only when the automatic setting of the X-axis display range (set with SCPI.DISPlay.PS(Ch).TRACe(Tr).X.SCALe.AUTO is set to OFF.

Variable

Parameter	*Value*
Description	Left setting for X axis
Data Type	Double precision floating point type (Double)
Range	0 ~ 499.999999999G
Preset Value	1k
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.DISPlay.PS1.TRACe1.X.SCALe.LEFT = Var
Var = SCPI.DISPlay.PS1.TRACe1.X.SCALe.LEFT
```

Related Objects

SCPI.DISPlay.PS(Ch).TRACe(Tr).X.SCALe.AUTO

SCPI.DISPlay.PS(Ch).TRACe(Tr).X.SCALe.RIGHt

Equivalent Key

PS Menu: **Scale** > **X Axis** > **Left**

Equivalent SCPI Command

Syntax

:DISPlay:PS[1-1]:TRACe[1-1]:X[:SCALe]:LEFT {0 ~ 499.999999999G}

:DISPlay:PS[1-1]:TRACe[1-1]:X[:SCALe]:LEFT?

Query Response

{0 ~ 499.999999999G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PS1:TRACe1:X[:SCALe]:LEFT 0"
20 OUTPUT 717;":DISPlay:PS1:TRACe1:X[:SCALe]:LEFT?"
30 ENTER 717;A
```

SCPI.DISPlay.PS(Ch).TRACe(Tr).X.SCALe.RIGHt

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PS(Ch).TRACe(Tr).X.SCALe.RIGHt = *Value*

Value = SCPI.DISPlay.PS(Ch).TRACe(Tr).X.SCALe.RIGHt

Description

This command sets/gets right setting for X axis, for the selected trace *Tr* of the selected channel *Ch*.

NOTE This command is available only when the automatic setting of the X-axis display range (set with SCPI.DISPlay.PS(Ch).TRACe(Tr).X.SCALe.AUTO is set to OFF.

Variable

Parameter	*Value*
Description	Right setting for X axis
Data Type	Double precision floating point type (Double)
Range	10m ~ 500G
Preset Value	1M
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
Var= 1e5
SCPI.DISPlay.PS1.TRACe1.X.SCALe.RIGHt = Var
Var = SCPI.DISPlay.PS1.TRACe1.X.SCALe.RIGHt
```

Related Objects

SCPI.DISPlay.PS(Ch).TRACe(Tr).X.SCALe.AUTO

SCPI.DISPlay.PS(Ch).TRACe(Tr).X.SCALe.LEFT

Equivalent Key

PS Menu: **Scale** > **X Axis** > **Right**

Equivalent SCPI Command

Syntax

:DISPlay:PS[1-1]:TRACe[1-1]:X[:SCALe]:RIGHt {10m ~ 500G}

:DISPlay:PS[1-1]:TRACe[1-1]:X[:SCALe]:RIGHt?

Query Response

{10m ~ 500G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PS1:TRACe1:X[:SCALe]:RIGHt 1e5"
20 OUTPUT 717;":DISPlay:PS1:TRACe1:X[:SCALe]:RIGHt?"
30 ENTER 717;A
```

SCPI.DISPlay.PS(Ch).TRACe(Tr).Y.SCALe.AUTO

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.PS(Ch).TRACe(Tr).Y.SCALe.AUTO

Description

This command sets auto scale setting for Y axis, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.DISPlay.PS1.TRACe1.Y.SCALe.AUTO

Related Objects

SCPI.DISPlay.PS(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.PS(Ch).TRACe(Tr).Y.SCALe.RLEVel

SCPI.DISPlay.PS(Ch).TRACe(Tr).Y.SCALe.RPOSition

Equivalent Key

PS Menu: **Scale** > **Auto Scale**

Equivalent SCPI Command

Syntax

:DISPlay:PS[1-1]:TRACe[1-1]:Y[:SCALe]:AUTO

Example of use

```
10 OUTPUT 717;":DISPlay:PS1:TRACe1:Y[:SCALe]:AUTO"
```

SCPI.DISPlay.PS(Ch).TRACe(Tr).Y.SCALe.PDIVision

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PS(Ch).TRACe(Tr).Y.SCALe.PDIVision = *Value*

Value = SCPI.DISPlay.PS(Ch).TRACe(Tr).Y.SCALe.PDIVision

Description

This command sets/gets scale per division, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale per division
Data Type	Double precision floating point type (Double)
Range	1a ~ 10G
Preset Value	10
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 1e-12
SCPI.DISPlay.PS1.TRACe1.Y.SCALe.PDIVision = Var
Var = SCPI.DISPlay.PS1.TRACe1.Y.SCALe.PDIVision
```

Related Objects

SCPI.DISPlay.PS(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.PS(Ch).TRACe(Tr).Y.SCALe.RLEVel

SCPI.DISPlay.PS(Ch).TRACe(Tr).Y.SCALe.RPOSition

Equivalent Key

PS Menu: **Scale** > **Scale/Div**

Equivalent SCPI Command

Syntax

:DISPlay:PS[1-1]:TRACe[1-1]:Y[:SCALe]:PDIVision {1a ~ 10G}

:DISPlay:PS[1-1]:TRACe[1-1]:Y[:SCALe]:PDIVision?

Query Response

{1a ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PS1:TRACe1:Y[:SCALe]:PDIVision 1e-12"
20 OUTPUT 717;":DISPlay:PS1:TRACe1:Y[:SCALe]:PDIVision?"
30 ENTER 717;A
```

SCPI.DISPlay.PS(Ch).TRACe(Tr).Y.SCALe.RLEVel

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PS(Ch).TRACe(Tr).Y.SCALe.RLEVel = *Value*

Value = SCPI.DISPlay.PS(Ch).TRACe(Tr).Y.SCALe.RLEVel

Description

This command sets/gets scale reference level, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale reference level
Data Type	Double precision floating point type (Double)
Range	-50G ~ 50G
Preset Value	-20
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -50e9
SCPI.DISPlay.PS1.TRACe1.Y.SCALe.RLEVel = Var
Var = SCPI.DISPlay.PS1.TRACe1.Y.SCALe.RLEVel
```

Related Objects

SCPI.DISPlay.PS(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.PS(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.PS(Ch).TRACe(Tr).Y.SCALe.RPOSition

Equivalent Key

PS Menu: **Scale** > **Reference Value**

PS Menu: **Scale** > **Marker** > **Reference**

Equivalent SCPI Command

Syntax

:DISPlay:PS[1-1]:TRACe[1-1]:Y[:SCALe]:RLEVel {-50G ~ 50G}

:DISPlay:PS[1-1]:TRACe[1-1]:Y[:SCALe]:RLEVel?

Query Response

{-50G ~ 50G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PS1:TRACe1:Y[:SCALe]:RLEVel -50e9"
20 OUTPUT 717;":DISPlay:PS1:TRACe1:Y[:SCALe]:RLEVel?"
30 ENTER 717;A
```

SCPI.DISPlay.PS(Ch).TRACe(Tr).Y.SCALe.RPOSition

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PS(Ch).TRACe(Tr).Y.SCALe.RPOSition = *Value*

Value = SCPI.DISPlay.PS(Ch).TRACe(Tr).Y.SCALe.RPOSition

Description

This command sets/gets scale reference position, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale reference position
Data Type	Long integer type (Long)
Range	0 ~ 30
Preset Value	10
Unit	Div
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.DISPlay.PS1.TRACe1.Y.SCALe.RPOSition = Var
Var = SCPI.DISPlay.PS1.TRACe1.Y.SCALe.RPOSition
```

Related Objects

SCPI.DISPlay.PS(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.PS(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.PS(Ch).TRACe(Tr).Y.SCALe.RLEVel

Equivalent Key

PS Menu: **Scale** > **Reference Position**

Equivalent SCPI Command

Syntax

:DISPlay:PS[1-1]:TRACe[1-1]:Y[:SCALe]:RPOSition {0 ~ 30}

:DISPlay:PS[1-1]:TRACe[1-1]:Y[:SCALe]:RPOSition?

Query Response

{0 ~ 30} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PS1:TRACe1:Y[:SCALe]:RPOSition 0 "
20 OUTPUT 717;":DISPlay:PS1:TRACe1:Y[:SCALe]:RPOSition?"
30 ENTER 717;A
```

SCPI.DISPlay.PS(Ch).Y.SCALe.DIVisions

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.PS(Ch).Y.SCALe.DIVisions = *Value*

Value = SCPI.DISPlay.PS(Ch).Y.SCALe.DIVisions

Description

This command sets/gets number of Y division, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Number of Y division
Data Type	Long integer type (Long)
Range	4 ~ 30
Preset Value	10
Unit	-
Resolution	2
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 4
SCPI.DISPlay.PS1.Y.SCALe.DIVisions = Var
Var = SCPI.DISPlay.PS1.Y.SCALe.DIVisions
```

Equivalent Key

PS Menu: **Scale** > **Divisions**

Equivalent SCPI Command

Syntax

:DISPlay:PS[1-1]:Y[:SCALe]:DIVisions {4 ~ 30}

:DISPlay:PS[1-1]:Y[:SCALe]:DIVisions?

Query Response

{4 ~ 30} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:PS1:Y[:SCALe]:DIVisions 4 "
20 OUTPUT 717;":DISPlay:PS1:Y[:SCALe]:DIVisions ?"
30 ENTER 717;A
```

SCPI.DISPlay.SKEY.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.SKEY.STATe = *Value*

Value = SCPI.DISPlay.SKEY.STATe

Description

This command Show/Hide soft key.

Variable

Parameter	*Value*
Description	Show/Hide soft key
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.SKEY.STATe = Var
Var = SCPI.DISPlay.SKEY.STATe
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:SKEY:STATe {ON|OFF|1|0}

:DISPlay:SKEY:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:SKEY:STATe 1"
20 OUTPUT 717;":DISPlay:SKEY:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.SP(Ch).ALLTrace.PERSistence.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.SP(Ch).ALLTrace.PERSistence.CLEar

Description

This command clears all stored traces, for the selected channel *Ch*.

Examples

SCPI.DISPlay.SP1.ALLTrace.PERSistence.CLEar

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:ALLTrace:PERSistence:CLEar

Example of use

10 OUTPUT 717;":DISPlay:SP1:ALLTrace:PERSistence:CLEar"

SCPI.DISPlay.SP(Ch).ANNotation.MARKer.POSition

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.SP(Ch).ANNotation.MARKer.POSition = *Value*

Value = SCPI.DISPlay.SP(Ch).ANNotation.MARKer.POSition

Description

This command sets/gets Marker annotation position, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker annotation position
Data Type	Character string type (String)
Range	LEFT|RIGHt
Preset Value	LEFT
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "LEFT"
SCPI.DISPlay.SP1.ANNotation.MARKer.POSition = Var
Var = SCPI.DISPlay.SP1.ANNotation.MARKer.POSition
```

Related Objects

SCPI.DISPlay.SP(Ch).ANNotation.MEASurement.STATe

Equivalent Key

SP Menu: **Display** > **Marker Information**

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:ANNotation:MARKer:POSition {LEFT|RIGHt}

:DISPlay:SP[1-1]:ANNotation:MARKer:POSition?

Query Response

{LEFT|RIGHt} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:SP1:ANNotation:MARKer:POSition LEFT"
20 OUTPUT 717;":DISPlay:SP1:ANNotation:MARKer:POSition ?"
30 ENTER 717;A$
```

SCPI.DISPlay.SP(Ch).ANNotation.MEASurement.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.SP(Ch).ANNotation.MEASurement.STATe = *Value*

Value = SCPI.DISPlay.SP(Ch).ANNotation.MEASurement.STATe

Description

This command Show/Hide measurement setting, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/Hide measurement setting
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.SP1.ANNotation.MEASurement.STATe = Var
Var = SCPI.DISPlay.SP1.ANNotation.MEASurement.STATe
```

Related Objects

SCPI.DISPlay.SP(Ch).ANNotation.MARKer.POSition

Equivalent Key

SP Menu: **Display** > **Meas Condition**

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:ANNotation:MEASurement:STATe {ON|OFF|1|0}

:DISPlay:SP[1-1]:ANNotation:MEASurement:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:SP1:ANNotation:MEASurement:STATe 1"
20 OUTPUT 717;":DISPlay:SP1:ANNotation:MEASurement:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.SP(Ch).GRATicule.AXIS.Y.RELative

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.SP(Ch).GRATicule.AXIS.Y.RELative = *Value*

Value = SCPI.DISPlay.SP(Ch).GRATicule.AXIS.Y.RELative

Description

This command sets/gets force graticule label notation relative. If OFF, absolute notation is used if possible, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Force graticule label notation
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.SP1.GRATicule.AXIS.Y.RELative = Var
Var = SCPI.DISPlay.SP1.GRATicule.AXIS.Y.RELative
```

Related Objects

SCPI.DISPlay.SP(Ch).GRATicule.AXIS.Y.STATe

Equivalent Key

SP Menu: **Display** > **Relative Y-Scale**

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:GRATicule:AXIS:Y:RELative {ON|OFF|1|0}

:DISPlay:SP[1-1]:GRATicule:AXIS:Y:RELative?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:SP1:GRATicule:AXIS:Y:RELative 1"
20 OUTPUT 717;":DISPlay:SP1:GRATicule:AXIS:Y:RELative ?"
30 ENTER 717;A
```

SCPI.DISPlay.SP(Ch).GRATicule.AXIS.Y.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.SP(Ch).GRATicule.AXIS.Y.STATe = *Value*

Value = SCPI.DISPlay.SP(Ch).GRATicule.AXIS.Y.STATe

Description

This command Show/Hide Y graticule label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/Hide Y graticule label
Data Type	Character string type (String)
Range	OFF|SHORt|MIDDle|LONG
Preset Value	SHORt
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.DISPlay.SP1.GRATicule.AXIS.Y.STATe = Var
Var = SCPI.DISPlay.SP1.GRATicule.AXIS.Y.STATe
```

Related Objects

SCPI.DISPlay.SP(Ch).GRATicule.AXIS.Y.RELative

Equivalent Key

SP Menu: **Display** > **Y # of Digits**

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:GRATicule:AXIS:Y:STATe {OFF|SHORt|MIDDle|LONG}

:DISPlay:SP[1-1]:GRATicule:AXIS:Y:STATe?

Query Response

{OFF|SHORt|MIDDle|LONG} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:SP1:GRATicule:AXIS:Y:STATe OFF"
20 OUTPUT 717;":DISPlay:SP1:GRATicule:AXIS:Y:STATe ?"
30 ENTER 717;A$
```

SCPI.DISPlay.SP(Ch).LABel.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.SP(Ch).LABel.DATA = *Value*

Value = SCPI.DISPlay.SP(Ch).LABel.DATA

Description

This command sets/gets Window Title Label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Window Title Label
Data Type	Character string type (String)
Range	254 chars
Preset Value	""
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
SCPI.DISPlay.SP1.LABel.DATA = Var
Var = SCPI.DISPlay.SP1.LABel.DATA
```

Related Objects

SCPI.DISPlay.SP(Ch).LABel.STATe

Equivalent Key

SP Menu: **Display** > **Edit Title Label**

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:LABel:DATA

:DISPlay:SP[1-1]:LABel:DATA?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:SP1:LABel:DATA Sample_Text"
20 OUTPUT 717;":DISPlay:SP1:LABel:DATA ?"
30 ENTER 717;A$
```

SCPI.DISPlay.SP(Ch).LABel.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.SP(Ch).LABel.STATe = *Value*

Value = SCPI.DISPlay.SP(Ch).LABel.STATe

Description

This command Show/Hide Window Title Label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/Hide Window Title Label
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.SP1.LABel.STATe = Var
Var = SCPI.DISPlay.SP1.LABel.STATe
```

Related Objects

SCPI.DISPlay.SP(Ch).LABel.DATA

Equivalent Key

SP Menu: **Display** > **Title Label**

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:LABel:STATe {ON|OFF|1|0}

:DISPlay:SP[1-1]:LABel:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:SP1:LABel:STATe 1"
20 OUTPUT 717;":DISPlay:SP1:LABel:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.SP(Ch).LIMit.FSIGn

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.SP(Ch).LIMit.FSIGn = *Value*

Value = SCPI.DISPlay.SP(Ch).LIMit.FSIGn

Description

This command show/hide fail sign, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	show/hide fail sign
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.SP1.LIMit.FSIGn = Var
Var = SCPI.DISPlay.SP1.LIMit.FSIGn
```

Equivalent Key

SP Menu: **Display** > **Limit Test** > **Fail Sign**

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:LIMit:FSIGn {ON|OFF|1|0}

:DISPlay:SP[1-1]:LIMit:FSIGn?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:SP1:LIMit:FSIGn 1"
20 OUTPUT 717;":DISPlay:SP1:LIMit:FSIGn ?"
30 ENTER 717;A
```

SCPI.DISPlay.SP(Ch).MAXimize

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.SP(Ch).MAXimize = *Value*

Value = SCPI.DISPlay.SP(Ch).MAXimize

Description

This command sets/gets maximize active trace, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Maximize active trace
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.SP1.MAXimize = Var
Var = SCPI.DISPlay.SP1.MAXimize
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:MAXimize {ON|OFF|1|0}

:DISPlay:SP[1-1]:MAXimize?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:SP1:MAXimize 1"
20 OUTPUT 717;":DISPlay:SP1:MAXimize ?"
30 ENTER 717;A
```

SCPI.DISPlay.SP(Ch).STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.SP(Ch).STATe = *Value*

Value = SCPI.DISPlay.SP(Ch).STATe

Description

This command sets/gets measurement display on/off. At least one meas window must be turned on, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Measurement display (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	60, "Failed to hide window" 100, "Option not installed"

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.SP1.STATe = Var
Var = SCPI.DISPlay.SP1.STATe
```

Equivalent Key

PN Menu: **Measurement View** > **Show Window** > **Spectrum Monitor**

SP Menu: **Measurement View** > **Show Window** > **Spectrum Monitor**

FP Menu: **Measurement View** > **Show Window** > **Spectrum Monitor**

TR Menu: **Measurement View** > **Show Window** > **Spectrum Monitor**

AM Menu: **Measurement View** > **Show Window** > **Spectrum Monitor**

BB Menu: **Measurement View** > **Show Window** > **Spectrum Monitor**

USER Menu: **Measurement View** > **Show Window** > **Spectrum Monitor**

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:STATe {ON|OFF|1|0}

:DISPlay:SP[1-1]:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:SP1:STATe 1"
20 OUTPUT 717;":DISPlay:SP1:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.SP(Ch).TABLe.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.SP(Ch).TABLe.STATe = *Value*

Value = SCPI.DISPlay.SP(Ch).TABLe.STATe

Description

This command show/hide text area, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/hide text area
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.SP1.TABLe.STATe = Var
Var = SCPI.DISPlay.SP1.TABLe.STATe
```

Equivalent Key

SP Menu: **Marker** > **Marker List**

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:TABLe[:STATe] {ON|OFF|1|0}

:DISPlay:SP[1-1]:TABLe[:STATe]?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:SP1:TABLe[:STATe] 1"
20 OUTPUT 717;":DISPlay:SP1:TABLe[:STATe] ?"
30 ENTER 717;A
```

SCPI.DISPlay.SP(Ch).TRACe(Tr).LABel.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.SP(Ch).TRACe(Tr).LABel.DATA = *Value*

Value = SCPI.DISPlay.SP(Ch).TRACe(Tr).LABel.DATA

Description

This command sets/gets Trace Title Label, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace Title Label
Data Type	Character string type (String)
Range	254 chars
Preset Value	"Spectrum"
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
SCPI.DISPlay.SP1.TRACe1.LABel.DATA = Var
Var = SCPI.DISPlay.SP1.TRACe1.LABel.DATA
```

Equivalent Key

SP Menu: **Trace View** > **Trace Label**

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:TRACe[1-1]:LABel:DATA

:DISPlay:SP[1-1]:TRACe[1-1]:LABel:DATA?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:SP1:TRACe1:LABel:DATA Sample_Text"
20 OUTPUT 717;":DISPlay:SP1:TRACe1:LABel:DATA ?"
30 ENTER 717;A$
```

SCPI.DISPlay.SP(Ch).TRACe(Tr).LIMit.LINE

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.SP(Ch).TRACe(Tr).LIMit.LINE = *Value*

Value = SCPI.DISPlay.SP(Ch).TRACe(Tr).LIMit.LINE

Description

This command show/hide limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/hide limit line
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.SP1.TRACe1.LIMit.LINE = Var
Var = SCPI.DISPlay.SP1.TRACe1.LIMit.LINE
```

Equivalent Key

SP Menu: **Display** > **Limit Test** > **Limit Line**

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:TRACe[1-1]:LIMit:LINE {ON|OFF|1|0}

:DISPlay:SP[1-1]:TRACe[1-1]:LIMit:LINE?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:SP1:TRACe1:LIMit:LINE 1"
20 OUTPUT 717;":DISPlay:SP1:TRACe1:LIMit:LINE ?"
30 ENTER 717;A
```

SCPI.DISPlay.SP(Ch).TRACe(Tr).MODE

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.SP(Ch).TRACe(Tr).MODE = *Value*

Value = SCPI.DISPlay.SP(Ch).TRACe(Tr).MODE

Description

This command sets/gets data and/or memory trace, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Data and/or memory trace
Data Type	Character string type (String)
Range	OFF|DATA|MEMory|BOTH
Preset Value	DATA
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.DISPlay.SP1.TRACe1.MODE = Var
Var = SCPI.DISPlay.SP1.TRACe1.MODE
```

Equivalent Key

SP Menu: **Trace View** > **Display Trace**

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:TRACe[1-1]:MODE {OFF|DATA|MEMory|BOTH}

:DISPlay:SP[1-1]:TRACe[1-1]:MODE?

Query Response

{OFF|DATA|MEMory|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:SP1:TRACe1:MODE OFF"
20 OUTPUT 717;":DISPlay:SP1:TRACe1:MODE ?"
30 ENTER 717;A$
```

SCPI.DISPlay.SP(Ch).TRACe(Tr).PERSistence.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.SP(Ch).TRACe(Tr).PERSistence.CLEar

Description

This command clears stored traces, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.DISPlay.SP1.TRACe1.PERSistence.CLEar

Related Objects

SCPI.DISPlay.SP(Ch).TRACe(Tr).PERSistence.STATe

Equivalent Key

SP Menu: **Trace View** > **Persistence** > **Clear Persistent Data**

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:TRACe[1-1]:PERSistence:CLEar

Example of use

```
10 OUTPUT 717;":DISPlay:SP1:TRACe1:PERSistence:CLEar"
```

SCPI.DISPlay.SP(Ch).TRACe(Tr).PERSistence.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.SP(Ch).TRACe(Tr).PERSistence.STATe = *Value*

Value = SCPI.DISPlay.SP(Ch).TRACe(Tr).PERSistence.STATe

Description

This command sets/gets trace storage mode, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace storage mode
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.SP1.TRACe1.PERSistence.STATe = Var
Var = SCPI.DISPlay.SP1.TRACe1.PERSistence.STATe
```

Related Objects

SCPI.DISPlay.SP(Ch).TRACe(Tr).PERSistence.CLEar

Equivalent Key

SP Menu: **Trace View** > **Persistence** > **Persistence Mode**

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:TRACe[1-1]:PERSistence:STATe {ON|OFF|1|0}

:DISPlay:SP[1-1]:TRACe[1-1]:PERSistence:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:SP1:TRACe1:PERSistence:STATe 1"
20 OUTPUT 717;":DISPlay:SP1:TRACe1:PERSistence:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.SP(Ch).TRACe(Tr).X.SCALe.AUTO

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.SP(Ch).TRACe(Tr).X.SCALe.AUTO = *Value*

Value = SCPI.DISPlay.SP(Ch).TRACe(Tr).X.SCALe.AUTO

Description

This command sets/gets autoscale setting for X axis, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Autoscale setting for X axis
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.SP1.TRACe1.X.SCALe.AUTO = Var
Var = SCPI.DISPlay.SP1.TRACe1.X.SCALe.AUTO
```

Related Objects

SCPI.DISPlay.SP(Ch).TRACe(Tr).X.SCALe.LEFT

SCPI.DISPlay.SP(Ch).TRACe(Tr).X.SCALe.RIGHt

Equivalent Key

SP Menu: **Scale** > **X Axis** > **Auto**

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:TRACe[1-1]:X[:SCALe]:AUTO {ON|OFF|1|0}

:DISPlay:SP[1-1]:TRACe[1-1]:X[:SCALe]:AUTO?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:SP1:TRACe1:X[:SCALe]:AUTO 1"
20 OUTPUT 717;":DISPlay:SP1:TRACe1:X[:SCALe]:AUTO ?"
30 ENTER 717;A
```

SCPI.DISPlay.SP(Ch).TRACe(Tr).X.SCALe.LEFT

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.SP(Ch).TRACe(Tr).X.SCALe.LEFT = *Value*

Value = SCPI.DISPlay.SP(Ch).TRACe(Tr).X.SCALe.LEFT

Description

This command sets/gets left setting for X axis, for the selected trace *Tr* of the selected channel *Ch*.

NOTE This command is available only when the automatic setting of the X-axis display range (set with SCPI.DISPlay.SP(Ch).TRACe(Tr).X.SCALe.AUTO) is set to OFF.

Variable

Parameter	*Value*
Description	Left setting for X axis
Data Type	Double precision floating point type (Double)
Range	0 ~ 499.999999999G
Preset Value	992.5M
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.DISPlay.SP1.TRACe1.X.SCALe.LEFT = Var
Var = SCPI.DISPlay.SP1.TRACe1.X.SCALe.LEFT
```

Related Objects

SCPI.DISPlay.SP(Ch).TRACe(Tr).X.SCALe.AUTO

SCPI.DISPlay.SP(Ch).TRACe(Tr).X.SCALe.RIGHt

Equivalent Key

SP Menu: **Scale** > **X Axis** > **Left**

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:TRACe[1-1]:X[:SCALe]:LEFT {0 ~ 499.999999999G}

:DISPlay:SP[1-1]:TRACe[1-1]:X[:SCALe]:LEFT?

Query Response

{0 ~ 499.999999999G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:SP1:TRACe1:X[:SCALe]:LEFT 0"
20 OUTPUT 717;":DISPlay:SP1:TRACe1:X[:SCALe]:LEFT ?"
30 ENTER 717;A
```

SCPI.DISPlay.SP(Ch).TRACe(Tr).X.SCALe.RIGHt

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.SP(Ch).TRACe(Tr).X.SCALe.RIGHt = *Value*

Value = SCPI.DISPlay.SP(Ch).TRACe(Tr).X.SCALe.RIGHt

Description

This command sets/gets right setting for X axis, for the selected trace *Tr* of the selected channel *Ch*.

NOTE This command is available only when the automatic setting of the X-axis display range (set with SCPI.DISPlay.SP(Ch).TRACe(Tr).X.SCALe.AUTO) is set to OFF.

Variable

Parameter	*Value*
Description	Right setting for X axis
Data Type	Double precision floating point type (Double)
Range	100m ~ 500G
Preset Value	1.0075G
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
Var= 0.1
SCPI.DISPlay.SP1.TRACe1.X.SCALe.RIGHt = Var
Var = SCPI.DISPlay.SP1.TRACe1.X.SCALe.RIGHt
```

Related Objects

SCPI.DISPlay.SP(Ch).TRACe(Tr).X.SCALe.AUTO

SCPI.DISPlay.SP(Ch).TRACe(Tr).X.SCALe.LEFT

Equivalent Key

SP Menu: **Scale** > **X Axis** > **Right**

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:TRACe[1-1]:X[:SCALe]:RIGHt {100m ~ 500G}

:DISPlay:SP[1-1]:TRACe[1-1]:X[:SCALe]:RIGHt?

Query Response

{100m ~ 500G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:SP1:TRACe1:X[:SCALe]:RIGHt 0.1"
20 OUTPUT 717;":DISPlay:SP1:TRACe1:X[:SCALe]:RIGHt ?"
30 ENTER 717;A
```

SCPI.DISPlay.SP(Ch).TRACe(Tr).Y.SCALe.AUTO

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.SP(Ch).TRACe(Tr).Y.SCALe.AUTO

Description

This command sets auto scale setting for Y axis, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.DISPlay.SP1.TRACe1.Y.SCALe.AUTO

Related Objects

SCPI.DISPlay.SP(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.SP(Ch).TRACe(Tr).Y.SCALe.RLEVel

SCPI.DISPlay.SP(Ch).TRACe(Tr).Y.SCALe.RPOSition

Equivalent Key

SP Menu: **Scale** > **Auto Scale**

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:TRACe[1-1]:Y[:SCALe]:AUTO

Example of use

```
10 OUTPUT 717;":DISPlay:SP1:TRACe1:Y[:SCALe]:AUTO"
```

SCPI.DISPlay.SP(Ch).TRACe(Tr).Y.SCALe.PDIVision

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.SP(Ch).TRACe(Tr).Y.SCALe.PDIVision = *Value*

Value = SCPI.DISPlay.SP(Ch).TRACe(Tr).Y.SCALe.PDIVision

Description

This command sets/gets scale per division, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale per division
Data Type	Double precision floating point type (Double)
Range	1a ~ 10G
Preset Value	10
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0.000000000001
SCPI.DISPlay.SP1.TRACe1.Y.SCALe.PDIVision = Var
Var = SCPI.DISPlay.SP1.TRACe1.Y.SCALe.PDIVision
```

Related Objects

SCPI.DISPlay.SP(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.SP(Ch).TRACe(Tr).Y.SCALe.RLEVel

SCPI.DISPlay.SP(Ch).TRACe(Tr).Y.SCALe.RPOSition

Equivalent Key

SP Menu: **Scale** > **Scale/Div**

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:TRACe[1-1]:Y[:SCALe]:PDIVision {1a ~ 10G}

:DISPlay:SP[1-1]:TRACe[1-1]:Y[:SCALe]:PDIVision?

Query Response

{1a ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:SP1:TRACe1:Y[:SCALe]:PDIVision 0.000000000001"
20 OUTPUT 717;":DISPlay:SP1:TRACe1:Y[:SCALe]:PDIVision ?"
30 ENTER 717;A
```

SCPI.DISPlay.SP(Ch).TRACe(Tr).Y.SCALe.RLEVel

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.SP(Ch).TRACe(Tr).Y.SCALe.RLEVel = *Value*

Value = SCPI.DISPlay.SP(Ch).TRACe(Tr).Y.SCALe.RLEVel

Description

This command sets/gets scale reference level, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale reference level
Data Type	Double precision floating point type (Double)
Range	-50G ~ 50G
Preset Value	10
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -50000000000
SCPI.DISPlay.SP1.TRACe1.Y.SCALe.RLEVel = Var
Var = SCPI.DISPlay.SP1.TRACe1.Y.SCALe.RLEVel
```

Related Objects

SCPI.DISPlay.SP(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.SP(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.SP(Ch).TRACe(Tr).Y.SCALe.RPOSition

Equivalent Key

SP Menu: **Scale** > **Reference Value**

SP Menu: **Scale** > **Marker** > **Reference**

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:TRACe[1-1]:Y[:SCALe]:RLEVel {-50G ~ 50G}

:DISPlay:SP[1-1]:TRACe[1-1]:Y[:SCALe]:RLEVel?

Query Response

{-50G ~ 50G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:SP1:TRACe1:Y[:SCALe]:RLEVel -50000000000"
20 OUTPUT 717;":DISPlay:SP1:TRACe1:Y[:SCALe]:RLEVel ?"
30 ENTER 717;A
```

SCPI.DISPlay.SP(Ch).TRACe(Tr).Y.SCALe.RPOSition

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.SP(Ch).TRACe(Tr).Y.SCALe.RPOSition = *Value*

Value = SCPI.DISPlay.SP(Ch).TRACe(Tr).Y.SCALe.RPOSition

Description

This command sets/gets scale reference position, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale reference position
Data Type	Long integer type (Long)
Range	0 ~ 30
Preset Value	10
Unit	Div
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.DISPlay.SP1.TRACe1.Y.SCALe.RPOSition = Var
Var = SCPI.DISPlay.SP1.TRACe1.Y.SCALe.RPOSition
```

Related Objects

SCPI.DISPlay.SP(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.SP(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.SP(Ch).TRACe(Tr).Y.SCALe.RLEVel

Equivalent Key

SP Menu: **Scale** > **Reference Position**

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:TRACe[1-1]:Y[:SCALe]:RPOSition {0 ~ 30}

:DISPlay:SP[1-1]:TRACe[1-1]:Y[:SCALe]:RPOSition?

Query Response

{0 ~ 30} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:SP1:TRACe1:Y[:SCALe]:RPOSition 0 "
20 OUTPUT 717;":DISPlay:SP1:TRACe1:Y[:SCALe]:RPOSition ?"
30 ENTER 717;A
```

SCPI.DISPlay.SP(Ch).Y.SCALe.DIVisions

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.SP(Ch).Y.SCALe.DIVisions = *Value*

Value = SCPI.DISPlay.SP(Ch).Y.SCALe.DIVisions

Description

This command sets/gets number of Y division, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Number of Y division
Data Type	Long integer type (Long)
Range	4 ~ 30
Preset Value	10
Unit	-
Resolution	2
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 4
SCPI.DISPlay.SP1.Y.SCALe.DIVisions = Var
Var = SCPI.DISPlay.SP1.Y.SCALe.DIVisions
```

Equivalent Key

SP Menu: **Scale** > **Divisions**

Equivalent SCPI Command

Syntax

:DISPlay:SP[1-1]:Y[:SCALe]:DIVisions {4 ~ 30}

:DISPlay:SP[1-1]:Y[:SCALe]:DIVisions?

Query Response

{4 ~ 30} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:SP1:Y[:SCALe]:DIVisions 4 "
20 OUTPUT 717;":DISPlay:SP1:Y[:SCALe]:DIVisions ?"
30 ENTER 717;A
```

SCPI.DISPlay.TR(Ch).ALLTrace.PERSistence.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.TR(Ch).ALLTrace.PERSistence.CLEar

Description

This command clears all stored traces, for the selected channel *Ch*.

Examples

SCPI.DISPlay.TR1.ALLTrace.PERSistence.CLEar

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:ALLTrace:PERSistence:CLEar

Example of use

10 OUTPUT 717;":DISPlay:TR1:ALLTrace:PERSistence:CLEar"

SCPI.DISPlay.TR(Ch).ALLTrace.Y.SCALe.AUTO

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.TR(Ch).ALLTrace.Y.SCALe.AUTO

Description

This command sets auto scale all, for the selected channel *Ch*.

Examples

SCPI.DISPlay.TR1.ALLTrace.Y.SCALe.AUTO

Equivalent Key

TR Menu: **Scale** > **Auto Scale All**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:ALLTrace:Y:SCALe:AUTO

Example of use

10 OUTPUT 717;":DISPlay:TR1:ALLTrace:Y:SCALe:AUTO"

SCPI.DISPlay.TR(Ch).ANNotation.MARKer.POSition

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.TR(Ch).ANNotation.MARKer.POSition = *Value*

Value = SCPI.DISPlay.TR(Ch).ANNotation.MARKer.POSition

Description

This command sets/gets Marker annotation position, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker annotation position
Data Type	Character string type (String)
Range	LEFT|RIGHt
Preset Value	LEFT
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "LEFT"
SCPI.DISPlay.TR1.ANNotation.MARKer.POSition = Var
Var = SCPI.DISPlay.TR1.ANNotation.MARKer.POSition
```

Related Objects

SCPI.DISPlay.TR(Ch).ANNotation.MEASurement.STATe

Equivalent Key

TR Menu: **Display** > **Marker Information**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:ANNotation:MARKer:POSition {LEFT|RIGHt}

:DISPlay:TR[1-1]:ANNotation:MARKer:POSition?

Query Response

{LEFT|RIGHt} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:TR1:ANNotation:MARKer:POSition LEFT"
20 OUTPUT 717;":DISPlay:TR1:ANNotation:MARKer:POSition ?"
30 ENTER 717;A$
```

SCPI.DISPlay.TR(Ch).ANNotation.MEASurement.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.TR(Ch).ANNotation.MEASurement.STATe = *Value*

Value = SCPI.DISPlay.TR(Ch).ANNotation.MEASurement.STATe

Description

This command Show/Hide measurement setting, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/Hide measurement setting
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.TR1.ANNotation.MEASurement.STATe = Var
Var = SCPI.DISPlay.TR1.ANNotation.MEASurement.STATe
```

Related Objects

SCPI.DISPlay.TR(Ch).ANNotation.MARKer.POSition

Equivalent Key

TR Menu: **Display** > **Meas Condition**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:ANNotation:MEASurement:STATe {ON|OFF|1|0}

:DISPlay:TR[1-1]:ANNotation:MEASurement:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:TR1:ANNotation:MEASurement:STATe 1"
20 OUTPUT 717;":DISPlay:TR1:ANNotation:MEASurement:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.TR(Ch).GRATicule.AXIS.Y.RELative

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.TR(Ch).GRATicule.AXIS.Y.RELative = *Value*

Value = SCPI.DISPlay.TR(Ch).GRATicule.AXIS.Y.RELative

Description

This command sets/gets force graticule label notation relative. If OFF, absolute notation is used if possible, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Graticule label notation relative
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.TR1.GRATicule.AXIS.Y.RELative = Var
Var = SCPI.DISPlay.TR1.GRATicule.AXIS.Y.RELative
```

Related Objects

SCPI.DISPlay.TR(Ch).GRATicule.AXIS.Y.STATe

Equivalent Key

TR Menu: **Display** > **Relative Y-Scale**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:GRATicule:AXIS:Y:RELative {ON|OFF|1|0}

:DISPlay:TR[1-1]:GRATicule:AXIS:Y:RELative?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:TR1:GRATicule:AXIS:Y:RELative 1"
20 OUTPUT 717;":DISPlay:TR1:GRATicule:AXIS:Y:RELative ?"
30 ENTER 717;A
```

SCPI.DISPlay.TR(Ch).GRATicule.AXIS.Y.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.TR(Ch).GRATicule.AXIS.Y.STATe = *Value*

Value = SCPI.DISPlay.TR(Ch).GRATicule.AXIS.Y.STATe

Description

This command Show/Hide Y graticule label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/Hide Y graticule label
Data Type	Character string type (String)
Range	OFF|SHORt|MIDDle|LONG
Preset Value	SHORt
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.DISPlay.TR1.GRATicule.AXIS.Y.STATe = Var
Var = SCPI.DISPlay.TR1.GRATicule.AXIS.Y.STATe
```

Related Objects

SCPI.DISPlay.TR(Ch).GRATicule.AXIS.Y.RELative

Equivalent Key

TR Menu: **Display** > **Y # of Digits**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:GRATicule:AXIS:Y:STATe {OFF|SHORt|MIDDle|LONG}

:DISPlay:TR[1-1]:GRATicule:AXIS:Y:STATe?

Query Response

{OFF|SHORt|MIDDle|LONG} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:TR1:GRATicule:AXIS:Y:STATe OFF"
20 OUTPUT 717;":DISPlay:TR1:GRATicule:AXIS:Y:STATe ?"
30 ENTER 717;A$
```

SCPI.DISPlay.TR(Ch).LABel.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.TR(Ch).LABel.DATA = *Value*

Value = SCPI.DISPlay.TR(Ch).LABel.DATA

Description

This command sets/gets Window Title Label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Window Title Label
Data Type	Character string type (String)
Range	254 chars
Preset Value	""
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
SCPI.DISPlay.TR1.LABel.DATA = Var
Var = SCPI.DISPlay.TR1.LABel.DATA
```

Related Objects

SCPI.DISPlay.TR(Ch).LABel.STATe

Equivalent Key

TR Menu: **Display** > **Edit Title Label**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:LABel:DATA

:DISPlay:TR[1-1]:LABel:DATA?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:TR1:LABel:DATA Sample_Text"
20 OUTPUT 717;":DISPlay:TR1:LABel:DATA ?"
30 ENTER 717;A$
```

SCPI.DISPlay.TR(Ch).LABel.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.TR(Ch).LABel.STATe = *Value*

Value = SCPI.DISPlay.TR(Ch).LABel.STATe

Description

This command Show/Hide Window Title Label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/Hide Window Title Label
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.TR1.LABel.STATe = Var
Var = SCPI.DISPlay.TR1.LABel.STATe
```

Equivalent Key

TR Menu: **Display** > **Title Label**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:LABel:STATe {ON|OFF|1|0}

:DISPlay:TR[1-1]:LABel:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:TR1:LABel:STATe 1"
20 OUTPUT 717;":DISPlay:TR1:LABel:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.TR(Ch).LIMit.FSIGn

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.TR(Ch).LIMit.FSIGn = *Value*

Value = SCPI.DISPlay.TR(Ch).LIMit.FSIGn

Description

This command show/hide fail sign, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	show/hide fail sign
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.TR1.LIMit.FSIGn = Var
Var = SCPI.DISPlay.TR1.LIMit.FSIGn
```

Equivalent Key

TR Menu: **Display** > **Limit Test** > **Fail Sign**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:LIMit:FSIGn {ON|OFF|1|0}

:DISPlay:TR[1-1]:LIMit:FSIGn?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:TR1:LIMit:FSIGn 1"
20 OUTPUT 717;":DISPlay:TR1:LIMit:FSIGn ?"
30 ENTER 717;A
```

SCPI.DISPlay.TR(Ch).MAXimize

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.TR(Ch).MAXimize = *Value*

Value = SCPI.DISPlay.TR(Ch).MAXimize

Description

This command sets/gets maximize active trace, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	maximize active trace
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.TR1.MAXimize = Var
Var = SCPI.DISPlay.TR1.MAXimize
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:MAXimize {ON|OFF|1|0}

:DISPlay:TR[1-1]:MAXimize?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:TR1:MAXimize 1"
20 OUTPUT 717;":DISPlay:TR1:MAXimize ?"
30 ENTER 717;A
```

SCPI.DISPlay.TR(Ch).PARameter

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.TR(Ch).PARameter = *Value*

Value = SCPI.DISPlay.TR(Ch).PARameter

Description

This command sets/gets window setting, for the selected channel *Ch*.

NOTE This function is available when measurement mode is Narrow/Narrow.

Variable

Parameter	*Value*
Description	Window setting
Data Type	Character string type (String)
Range	ALL|FREQuency|POWer|PHASe
Preset Value	ALL
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "FREQuency"
SCPI.SENSe.TR1.MMODe = "NN"
SCPI.DISPlay.TR1.PARameter = Var
Var = SCPI.DISPlay.TR1.PARameter
```

Equivalent Key

TR Menu: **Display** > **Parameters**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:PARameter {ALL|FREQuency|POWer|PHASe}

:DISPlay:TR[1-1]:PARameter?

Query Response

{ALL|FREQuency|POWer|PHASe} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:TR1:MMODe NN"
20 OUTPUT 717;":DISPlay:TR1:PARameter FREQuency"
30 OUTPUT 717;":DISPlay:TR1:PARameter?"
40 ENTER 717;A$
```

SCPI.DISPlay.TR(Ch).STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.TR(Ch).STATe = *Value*

Value = SCPI.DISPlay.TR(Ch).STATe

Description

This command sets/gets measurement display on/off. At least one measurement window must be turned on, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Measurement display (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	60, "Failed to hide window" 100, "Option not installed"

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.TR1.STATe = Var
Var = SCPI.DISPlay.TR1.STATe
```

Equivalent Key

PN Menu: **Measurement View** > **Show Window** > **Transient**

SP Menu: **Measurement View** > **Show Window** > **Transient**

FP Menu: **Measurement View** > **Show Window** > **Transient**

TR Menu: **Measurement View** > **Show Window** > **Transient**

AM Menu: **Measurement View** > **Show Window** > **Transient**

BB Menu: **Measurement View** > **Show Window** > **Transient**

USER Menu: **Measurement View** > **Show Window** > **Transient**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:STATe {ON|OFF|1|0}

:DISPlay:TR[1-1]:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:TR1:STATe 1"
20 OUTPUT 717;":DISPlay:TR1:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.TR(Ch).TABLe.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.TR(Ch).TABLe.STATe = *Value*

Value = SCPI.DISPlay.TR(Ch).TABLe.STATe

Description

This command show/hide text area, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	show/hide text area
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.TR1.TABLe.STATe = Var
Var = SCPI.DISPlay.TR1.TABLe.STATe
```

Equivalent Key

TR Menu: **Marker** > **Marker List**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:TABLe[:STATe] {ON|OFF|1|0}

:DISPlay:TR[1-1]:TABLe[:STATe]?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:TR1:TABLe[:STATe] 1"
20 OUTPUT 717;":DISPlay:TR1:TABLe[:STATe] ?"
30 ENTER 717;A
```

SCPI.DISPlay.TR(Ch).TRACe(Tr).LABel.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.TR(Ch).TRACe(Tr).LABel.DATA = *Value*

Value = SCPI.DISPlay.TR(Ch).TRACe(Tr).LABel.DATA

Description

This command sets/gets Trace Title Label, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace Title Label
Data Type	Character string type (String)
Range	254 chars
Preset Value	"WB Freq"
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
SCPI.DISPlay.TR1.TRACe4.LABel.DATA = Var
Var = SCPI.DISPlay.TR1.TRACe4.LABel.DATA
```

Equivalent Key

TR Menu: **Trace View** > **Trace Label**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:TRACe[1-8]:LABel:DATA

:DISPlay:TR[1-1]:TRACe[1-8]:LABel:DATA?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:TR1:TRACe4:LABel:DATA Sample_Text"
20 OUTPUT 717;":DISPlay:TR1:TRACe4:LABel:DATA ?"
30 ENTER 717;A$
```

SCPI.DISPlay.TR(Ch).TRACe(Tr).LIMit.LINE

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.TR(Ch).TRACe(Tr).LIMit.LINE = *Value*

Value = SCPI.DISPlay.TR(Ch).TRACe(Tr).LIMit.LINE

Description

This command show/hide limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	show/hide limit line
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.TR1.TRACe4.LIMit.LINE = Var
Var = SCPI.DISPlay.TR1.TRACe4.LIMit.LINE
```

Equivalent Key

TR Menu: **Display** > **Limit Test** > **Limit Line**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:TRACe[1-8]:LIMit:LINE {ON|OFF|1|0}

:DISPlay:TR[1-1]:TRACe[1-8]:LIMit:LINE?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:TR1:TRACe4:LIMit:LINE 1"
20 OUTPUT 717;":DISPlay:TR1:TRACe4:LIMit:LINE ?"
30 ENTER 717;A
```

SCPI.DISPlay.TR(Ch).TRACe(Tr).MODE

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.TR(Ch).TRACe(Tr).MODE = *Value*

Value = SCPI.DISPlay.TR(Ch).TRACe(Tr).MODE

Description

This command sets/gets data and/or memory trace, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Data and/or memory trace
Data Type	Character string type (String)
Range	OFF|DATA|MEMory|BOTH
Preset Value	DATA
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.DISPlay.TR1.TRACe4.MODE = Var
Var = SCPI.DISPlay.TR1.TRACe4.MODE
```

Equivalent Key

TR Menu: **Trace View** > **Display Trace**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:TRACe[1-8]:MODE {OFF|DATA|MEMory|BOTH}

:DISPlay:TR[1-1]:TRACe[1-8]:MODE?

Query Response

{OFF|DATA|MEMory|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:TR1:TRACe4:MODE OFF"
20 OUTPUT 717;":DISPlay:TR1:TRACe4:MODE ?"
30 ENTER 717;A$
```

SCPI.DISPlay.TR(Ch).TRACe(Tr).PERSistence.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.TR(Ch).TRACe(Tr).PERSistence.CLEar

Description

This command clears stored traces, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.DISPlay.TR1.TRACe4.PERSistence.CLEar

Related Objects

SCPI.DISPlay.TR(Ch).TRACe(Tr).PERSistence.STATe

Equivalent Key

TR Menu: **Trace View** > **Persistence** > **Clear Persistent Data**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:TRACe[1-8]:PERSistence:CLEar

Example of use

10 OUTPUT 717;":DISPlay:TR1:TRACe4:PERSistence:CLEar"

SCPI.DISPlay.TR(Ch).TRACe(Tr).PERSistence.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.TR(Ch).TRACe(Tr).PERSistence.STATe = *Value*

Value = SCPI.DISPlay.TR(Ch).TRACe(Tr).PERSistence.STATe

Description

This command sets/gets trace storage mode, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace storage mode
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.TR1.TRACe4.PERSistence.STATe = Var
Var = SCPI.DISPlay.TR1.TRACe4.PERSistence.STATe
```

Related Objects

SCPI.DISPlay.TR(Ch).TRACe(Tr).PERSistence.CLEar

Equivalent Key

TR Menu: **Trace View** > **Persistence** > **Persistence Mode**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:TRACe[1-8]:PERSistence:STATe {ON|OFF|1|0}

:DISPlay:TR[1-1]:TRACe[1-8]:PERSistence:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:TR1:TRACe4:PERSistence:STATe 1"
20 OUTPUT 717;":DISPlay:TR1:TRACe4:PERSistence:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.TR(Ch).TRACe(Tr).X.SCALe.AUTO

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.TR(Ch).TRACe(Tr).X.SCALe.AUTO = *Value*

Value = SCPI.DISPlay.TR(Ch).TRACe(Tr).X.SCALe.AUTO

Description

This command sets/gets autoscale setting for X axis, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Autoscale setting for X axis
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.TR1.TRACe4.X.SCALe.AUTO = Var
Var = SCPI.DISPlay.TR1.TRACe4.X.SCALe.AUTO
```

Related Objects

SCPI.DISPlay.TR(Ch).TRACe(Tr).X.SCALe.AUTO

SCPI.DISPlay.TR(Ch).TRACe(Tr).X.SCALe.LEFT

SCPI.DISPlay.TR(Ch).TRACe(Tr).X.SCALe.RIGHt

Equivalent Key

TR Menu: **Scale** > **X Axis** > **Auto**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:TRACe[1-8]:X[:SCALe]:AUTO {ON|OFF|1|0}

:DISPlay:TR[1-1]:TRACe[1-8]:X[:SCALe]:AUTO?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:TR1:TRACe4:X[:SCALe]:AUTO 1"
20 OUTPUT 717;":DISPlay:TR1:TRACe4:X[:SCALe]:AUTO ?"
30 ENTER 717;A
```

SCPI.DISPlay.TR(Ch).TRACe(Tr).X.SCALe.LEFT

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.TR(Ch).TRACe(Tr).X.SCALe.LEFT = *Value*

Value = SCPI.DISPlay.TR(Ch).TRACe(Tr).X.SCALe.LEFT

Description

This command sets/gets left setting for X axis, for the selected trace *Tr* of the selected channel *Ch*.

NOTE This command is available only when the automatic setting of the X-axis display range (set with SCPI.DISPlay.TR(Ch).TRACe(Tr).X.SCALe.AUTO) is set to OFF.

Variable

Parameter	*Value*
Description	Left setting for X axis
Data Type	Double precision floating point type (Double)
Range	-8 ~ 10.99999999
Preset Value	-50m
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.DISPlay.TR1.TRACe4.X.SCALe.LEFT = Var
Var = SCPI.DISPlay.TR1.TRACe4.X.SCALe.LEFT
```

Related Objects

SCPI.DISPlay.TR(Ch).TRACe(Tr).X.SCALe.AUTO

SCPI.DISPlay.TR(Ch).TRACe(Tr).X.SCALe.RIGHt

Equivalent Key

TR Menu: **Scale** > **X Axis** > **Left**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:TRACe[1-8]:X[:SCALe]:LEFT {-8 ~ 10.99999999}

:DISPlay:TR[1-1]:TRACe[1-8]:X[:SCALe]:LEFT?

Query Response

{-8 ~ 10.99999999} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:TR1:TRACe4:X[:SCALe]:LEFT 0"
20 OUTPUT 717;":DISPlay:TR1:TRACe4:X[:SCALe]:LEFT ?"
30 ENTER 717;A
```

SCPI.DISPlay.TR(Ch).TRACe(Tr).X.SCALe.RIGHt

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.TR(Ch).TRACe(Tr).X.SCALe.RIGHt = *Value*

Value = SCPI.DISPlay.TR(Ch).TRACe(Tr).X.SCALe.RIGHt

Description

This command sets/gets right setting for X axis, for the selected trace *Tr* of the selected channel *Ch*.

NOTE This command is available only when the automatic setting of the X-axis display range (set with SCPI.DISPlay.TR(Ch).TRACe(Tr).X.SCALe.AUTO) is set to OFF.

Variable

Parameter	*Value*
Description	Right setting for X axis
Data Type	Double precision floating point type (Double)
Range	-7.99999999 ~ 11
Preset Value	50m
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.DISPlay.TR1.TRACe4.X.SCALe.RIGHt = Var
Var = SCPI.DISPlay.TR1.TRACe4.X.SCALe.RIGHt
```

Related Objects

SCPI.DISPlay.TR(Ch).TRACe(Tr).X.SCALe.AUTO

SCPI.DISPlay.TR(Ch).TRACe(Tr).X.SCALe.LEFT

Equivalent Key

TR Menu: **Scale** > **X Axis** > **Right**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:TRACe[1-8]:X[:SCALe]:RIGHt {-7.99999999 ~ 11}

:DISPlay:TR[1-1]:TRACe[1-8]:X[:SCALe]:RIGHt?

Query Response

{-7.99999999 ~ 11} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:TR1:TRACe4:X[:SCALe]:RIGHt 0"
20 OUTPUT 717;":DISPlay:TR1:TRACe4:X[:SCALe]:RIGHt ?"
30 ENTER 717;A
```

SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.AUTO

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.AUTO

Description

This command sets auto scale setting for Y axis, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.DISPlay.TR1.TRACe4.Y.SCALe.AUTO

Related Objects

SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.RLEVel

SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.RPOSition

Equivalent Key

TR Menu: **Scale** > **Auto Scale**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:TRACe[1-8]:Y[:SCALe]:AUTO

Example of use

```
10 OUTPUT 717;":DISPlay:TR1:TRACe4:Y[:SCALe]:AUTO"
```

SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.PDIVision

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.PDIVision = *Value*

Value = SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.PDIVision

Description

This command sets/gets scale per division, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale per division
Data Type	Double precision floating point type (Double)
Range	1a ~ 10G
Preset Value	80M
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0.000000000001
SCPI.DISPlay.TR1.TRACe4.Y.SCALe.PDIVision = Var
Var = SCPI.DISPlay.TR1.TRACe4.Y.SCALe.PDIVision
```

Related Objects

SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.RLEVel

SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.RPOSition

Equivalent Key

TR Menu: **Scale** > **Scale/Div**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:TRACe[1-8]:Y[:SCALe]:PDIVision {1a ~ 10G}

:DISPlay:TR[1-1]:TRACe[1-8]:Y[:SCALe]:PDIVision?

Query Response

{1a ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:TR1:TRACe4:Y[:SCALe]:PDIVision 0.000000000001"
20 OUTPUT 717;":DISPlay:TR1:TRACe4:Y[:SCALe]:PDIVision ?"
30 ENTER 717;A
```

SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.RLEVel

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.RLEVel = *Value*

Value = SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.RLEVel

Description

This command sets/gets scale reference level, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale reference level
Data Type	Double precision floating point type (Double)
Range	-500G ~ 500G
Preset Value	800M
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -500000000000
SCPI.DISPlay.TR1.TRACe4.Y.SCALe.RLEVel = Var
Var = SCPI.DISPlay.TR1.TRACe4.Y.SCALe.RLEVel
```

Related Objects

SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.RPOSition

Equivalent Key

TR Menu: **Scale** > **Reference Value**

TR Menu: **Scale** > **Marker** > **Reference**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:TRACe[1-8]:Y[:SCALe]:RLEVel {-500G ~ 500G}

:DISPlay:TR[1-1]:TRACe[1-8]:Y[:SCALe]:RLEVel?

Query Response

{-500G ~ 500G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:TR1:TRACe4:Y[:SCALe]:RLEVel -500000000000"
20 OUTPUT 717;":DISPlay:TR1:TRACe4:Y[:SCALe]:RLEVel ?"
30 ENTER 717;A
```

SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.RPOSition

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.RPOSition = *Value*

Value = SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.RPOSition

Description

This command sets/gets scale reference position, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale reference position
Data Type	Long integer type (Long)
Range	0 ~ 30
Preset Value	5
Unit	Div
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.DISPlay.TR1.TRACe4.Y.SCALe.RPOSition = Var
Var = SCPI.DISPlay.TR1.TRACe4.Y.SCALe.RPOSition
```

Related Objects

SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.TR(Ch).TRACe(Tr).Y.SCALe.RLEVel

Equivalent Key

TR Menu: **Scale** > **Reference Position**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:TRACe[1-8]:Y[:SCALe]:RPOSition {0 ~ 30}

:DISPlay:TR[1-1]:TRACe[1-8]:Y[:SCALe]:RPOSition?

Query Response

{0 ~ 30} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:TR1:TRACe4:Y[:SCALe]:RPOSition 0 "
20 OUTPUT 717;":DISPlay:TR1:TRACe4:Y[:SCALe]:RPOSition ?"
30 ENTER 717;A
```

SCPI.DISPlay.TR(Ch).Y.SCALe.DIVisions

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.TR(Ch).Y.SCALe.DIVisions = *Value*

Value = SCPI.DISPlay.TR(Ch).Y.SCALe.DIVisions

Description

This command sets/gets number of Y division, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Number of Y division
Data Type	Long integer type (Long)
Range	4 ~ 30
Preset Value	10
Unit	-
Resolution	2
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 4
SCPI.DISPlay.TR1.Y.SCALe.DIVisions = Var
Var = SCPI.DISPlay.TR1.Y.SCALe.DIVisions
```

Equivalent Key

TR Menu: **Scale** > **Divisions**

Equivalent SCPI Command

Syntax

:DISPlay:TR[1-1]:Y[:SCALe]:DIVisions {4 ~ 30}

:DISPlay:TR[1-1]:Y[:SCALe]:DIVisions?

Query Response

{4 ~ 30} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:TR1:Y[:SCALe]:DIVisions 4 "
20 OUTPUT 717;":DISPlay:TR1:Y[:SCALe]:DIVisions ?"
30 ENTER 717;A
```

SCPI.DISPlay.UPDate.IMMediate

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.UPDate.IMMediate

Description

This command immediately updates the display.

Examples

SCPI.DISPlay.UPDate.IMMediate

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:UPDate:IMMediate

Example of use

10 OUTPUT 717;":DISPlay:UPDate:IMMediate"

SCPI.DISPlay.USER(Ch).ALLTrace.PERSistence.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.USER(Ch).ALLTrace.PERSistence.CLEar

Description

This command clear all stored traces, for the selected channel *Ch*.

Examples

SCPI.DISPlay.USER1.ALLTrace.PERSistence.CLEar

Equivalent Key

USER Menu: **Trace View** > **Persistence** > **Clear All Persistent Data**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:ALLTrace:PERSistence:CLEar

Example of use

10 OUTPUT 717;":DISPlay:USER1:ALLTrace:PERSistence:CLEar"

SCPI.DISPlay.USER(Ch).ALLTrace.Y.SCALe.AUTO

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.USER(Ch).ALLTrace.Y.SCALe.AUTO

Description

This command sets auto scale all, for the selected channel *Ch*.

Examples

SCPI.DISPlay.USER1.ALLTrace.Y.SCALe.AUTO

Equivalent Key

USER Menu: **Scale** > **Auto Scale All**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:ALLTrace:Y:SCALe:AUTO

Example of use

10 OUTPUT 717;":DISPlay:USER1:ALLTrace:Y:SCALe:AUTO"

SCPI.DISPlay.USER(Ch).ANNotation.MARKer.POSition

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).ANNotation.MARKer.POSition = *Value*

Value = SCPI.DISPlay.USER(Ch).ANNotation.MARKer.POSition

Description

This command sets/gets Marker annotation position, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Marker annotation position
Data Type	Character string type (String)
Range	LEFT|RIGHt
Preset Value	LEFT
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "LEFT"
SCPI.DISPlay.USER1.ANNotation.MARKer.POSition = Var
Var = SCPI.DISPlay.USER1.ANNotation.MARKer.POSition
```

Related Objects

SCPI.DISPlay.USER(Ch).ANNotation.MEASurement.STATe

Equivalent Key

USER Menu: **Display** > **Marker Information**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:ANNotation:MARKer:POSition {LEFT|RIGHt}

:DISPlay:USER[1-1]:ANNotation:MARKer:POSition?

Query Response

{LEFT|RIGHt} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:ANNotation:MARKer:POSition LEFT"
20 OUTPUT 717;":DISPlay:USER1:ANNotation:MARKer:POSition ?"
30 ENTER 717;A$
```

SCPI.DISPlay.USER(Ch).ANNotation.MEASurement.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).ANNotation.MEASurement.STATe = *Value*

Value = SCPI.DISPlay.USER(Ch).ANNotation.MEASurement.STATe

Description

This command Show/Hide measurement setting, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/Hide measurement setting
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.USER1.ANNotation.MEASurement.STATe = Var
Var = SCPI.DISPlay.USER1.ANNotation.MEASurement.STATe
```

Related Objects

SCPI.DISPlay.USER(Ch).ANNotation.MARKer.POSition

Equivalent Key

USER Menu: **Display** > **Meas Condition**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:ANNotation:MEASurement:STATe {ON|OFF|1|0}

:DISPlay:USER[1-1]:ANNotation:MEASurement:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:ANNotation:MEASurement:STATe 1"
20 OUTPUT 717;":DISPlay:USER1:ANNotation:MEASurement:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.USER(Ch).GRATicule.AXIS.Y.RELative

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).GRATicule.AXIS.Y.RELative = *Value*

Value = SCPI.DISPlay.USER(Ch).GRATicule.AXIS.Y.RELative

Description

This command sets/gets graticule label notation relative. If OFF, absolute notation is used if possible, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Graticule label notation relative
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.USER1.GRATicule.AXIS.Y.RELative = Var
Var = SCPI.DISPlay.USER1.GRATicule.AXIS.Y.RELative
```

Related Objects

SCPI.DISPlay.USER(Ch).GRATicule.AXIS.Y.STATe

Equivalent Key

USER Menu: **Display** > **Relative Y-Scale**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:GRATicule:AXIS:Y:RELative {ON|OFF|1|0}

:DISPlay:USER[1-1]:GRATicule:AXIS:Y:RELative?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:GRATicule:AXIS:Y:RELative 1"
20 OUTPUT 717;":DISPlay:USER1:GRATicule:AXIS:Y:RELative ?"
30 ENTER 717;A
```

SCPI.DISPlay.USER(Ch).GRATicule.AXIS.Y.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).GRATicule.AXIS.Y.STATe = *Value*

Value = SCPI.DISPlay.USER(Ch).GRATicule.AXIS.Y.STATe

Description

This command Show/Hide Y graticule label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/Hide Y graticule label
Data Type	Character string type (String)
Range	OFF|SHORt|MIDDle|LONG
Preset Value	SHORt
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.DISPlay.USER1.GRATicule.AXIS.Y.STATe = Var
Var = SCPI.DISPlay.USER1.GRATicule.AXIS.Y.STATe
```

Related Objects

SCPI.DISPlay.USER(Ch).GRATicule.AXIS.Y.RELative

Equivalent Key

USER Menu: **Display** > **Y # of Digits**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:GRATicule:AXIS:Y:STATe {OFF|SHORt|MIDDle|LONG}

:DISPlay:USER[1-1]:GRATicule:AXIS:Y:STATe?

Query Response

{OFF|SHORt|MIDDle|LONG} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:GRATicule:AXIS:Y:STATe OFF"
20 OUTPUT 717;":DISPlay:USER1:GRATicule:AXIS:Y:STATe ?"
30 ENTER 717;A$
```

SCPI.DISPlay.USER(Ch).LABel.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).LABel.DATA = *Value*

Value = SCPI.DISPlay.USER(Ch).LABel.DATA

Description

This command sets/gets Window Title Label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Window Title Label
Data Type	Character string type (String)
Range	254 chars
Preset Value	""
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
SCPI.DISPlay.USER1.LABel.DATA = Var
Var = SCPI.DISPlay.USER1.LABel.DATA
```

Related Objects

SCPI.DISPlay.USER(Ch).LABel.STATe

Equivalent Key

USER Menu: **Display** > **Edit Title Label**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:LABel:DATA

:DISPlay:USER[1-1]:LABel:DATA?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:LABel:DATA Sample_Text"
20 OUTPUT 717;":DISPlay:USER1:LABel:DATA ?"
30 ENTER 717;A$
```

SCPI.DISPlay.USER(Ch).LABel.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).LABel.STATe = *Value*

Value = SCPI.DISPlay.USER(Ch).LABel.STATe

Description

This command show/hide Window Title Label, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	show/hide Window Title Label
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.USER1.LABel.STATe = Var
Var = SCPI.DISPlay.USER1.LABel.STATe
```

Related Objects

SCPI.DISPlay.USER(Ch).LABel.DATA

Equivalent Key

USER Menu: **Display** > **Title Label**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:LABel:STATe {ON|OFF|1|0}

:DISPlay:USER[1-1]:LABel:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:LABel:STATe 1"
20 OUTPUT 717;":DISPlay:USER1:LABel:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.USER(Ch).LIMit.FSIGn

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).LIMit.FSIGn = *Value*

Value = SCPI.DISPlay.USER(Ch).LIMit.FSIGn

Description

This command show/hide fail sign, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	show/hide fail sign
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.USER1.LIMit.FSIGn = Var
Var = SCPI.DISPlay.USER1.LIMit.FSIGn
```

Equivalent Key

USER Menu: **Display** > **Limit Test** > **Fail Sign**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:LIMit:FSIGn {ON|OFF|1|0}

:DISPlay:USER[1-1]:LIMit:FSIGn?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:LIMit:FSIGn 1"
20 OUTPUT 717;":DISPlay:USER1:LIMit:FSIGn ?"
30 ENTER 717;A
```

SCPI.DISPlay.USER(Ch).MAXimize

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).MAXimize = *Value*

Value = SCPI.DISPlay.USER(Ch).MAXimize

Description

This command sets/gets maximize active trace, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Maximize active trace
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.USER1.MAXimize = Var
Var = SCPI.DISPlay.USER1.MAXimize
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:MAXimize {ON|OFF|1|0}

:DISPlay:USER[1-1]:MAXimize?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:MAXimize 1"
20 OUTPUT 717;":DISPlay:USER1:MAXimize ?"
30 ENTER 717;A
```

SCPI.DISPlay.USER(Ch).STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).STATe = *Value*

Value = SCPI.DISPlay.USER(Ch).STATe

Description

This command sets/gets measurement display on/off. At least one meas window must be turned on, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Measurement display (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	60, "Failed to hide window" 100, "Option not installed"

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.USER1.STATe = Var
Var = SCPI.DISPlay.USER1.STATe
```

Equivalent Key

PN Menu: **Measurement View** > **Show Window** > **User**

SP Menu: **Measurement View** > **Show Window** > **User**

FP Menu: **Measurement View** > **Show Window** > **User**

TR Menu: **Measurement View** > **Show Window** > **User**

AM Menu: **Measurement View** > **Show Window** > **User**

BB Menu: **Measurement View** > **Show Window** > **User**

USER Menu: **Measurement View** > **Show Window** > **User**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:STATe {ON|OFF|1|0}

:DISPlay:USER[1-1]:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:STATe 1"
20 OUTPUT 717;":DISPlay:USER1:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.USER(Ch).TABLe.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).TABLe.STATe = *Value*

Value = SCPI.DISPlay.USER(Ch).TABLe.STATe

Description

This command show/hide text area, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/hide text area
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.USER1.TABLe.STATe = Var
Var = SCPI.DISPlay.USER1.TABLe.STATe
```

Equivalent Key

USER Menu: **Marker** > **Marker List**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:TABLe[:STATe] {ON|OFF|1|0}

:DISPlay:USER[1-1]:TABLe[:STATe]?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:TABLe[:STATe] 1"
20 OUTPUT 717;":DISPlay:USER1:TABLe[:STATe] ?"
30 ENTER 717;A
```

SCPI.DISPlay.USER(Ch).TRACe(Tr).ANNotation.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).TRACe(Tr).ANNotation.DATA = *Value*

Value = SCPI.DISPlay.USER(Ch).TRACe(Tr).ANNotation.DATA

Description

This command sets/gets Trace Annotation, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace Annotation
Data Type	Character string type (String)
Range	254 chars
Preset Value	""
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
SCPI.DISPlay.USER1.TRACe8.ANNotation.DATA = Var
Var = SCPI.DISPlay.USER1.TRACe8.ANNotation.DATA
```

Equivalent Key

USER Menu: **Trace View** > **Trace Annotation**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:TRACe[1-8]:ANNotation:DATA

:DISPlay:USER[1-1]:TRACe[1-8]:ANNotation:DATA?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:TRACe8:ANNotation:DATA Sample_Text"
20 OUTPUT 717;":DISPlay:USER1:TRACe8:ANNotation:DATA ?"
30 ENTER 717;A$
```

SCPI.DISPlay.USER(Ch).TRACe(Tr).LABel.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).TRACe(Tr).LABel.DATA = *Value*

Value = SCPI.DISPlay.USER(Ch).TRACe(Tr).LABel.DATA

Description

This command sets/gets Trace Title Label, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace Title Label
Data Type	Character string type (String)
Range	254 chars
Preset Value	"Tr1"
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
SCPI.DISPlay.USER1.TRACe8.LABel.DATA = Var
Var = SCPI.DISPlay.USER1.TRACe8.LABel.DATA
```

Equivalent Key

USER Menu: **Trace View** > **Trace Label**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:TRACe[1-8]:LABel:DATA

:DISPlay:USER[1-1]:TRACe[1-8]:LABel:DATA?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:TRACe8:LABel:DATA Sample_Text"
20 OUTPUT 717;":DISPlay:USER1:TRACe8:LABel:DATA ?"
30 ENTER 717;A$
```

SCPI.DISPlay.USER(Ch).TRACe(Tr).LIMit.LINE

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).TRACe(Tr).LIMit.LINE = *Value*

Value = SCPI.DISPlay.USER(Ch).TRACe(Tr).LIMit.LINE

Description

This command show/hide limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show/hide limit line
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.USER1.TRACe8.LIMit.LINE = Var
Var = SCPI.DISPlay.USER1.TRACe8.LIMit.LINE
```

Equivalent Key

USER Menu: **Display** > **Limit Test** > **Limit Line**

Equivalent SCPI Command

Syntax

```
:DISPlay:USER[1-1]:TRACe[1-8]:LIMit:LINE {ON|OFF|1|0}
:DISPlay:USER[1-1]:TRACe[1-8]:LIMit:LINE?
```

Query Response

```
{ON|OFF|1|0} <newline><^END>
```

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:TRACe8:LIMit:LINE 1"
20 OUTPUT 717;":DISPlay:USER1:TRACe8:LIMit:LINE ?"
30 ENTER 717;A
```

SCPI.DISPlay.USER(Ch).TRACe(Tr).MODE

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).TRACe(Tr).MODE = *Value*

Value = SCPI.DISPlay.USER(Ch).TRACe(Tr).MODE

Description

This command show data and/or memory trace, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Show data and/or memory trace
Data Type	Character string type (String)
Range	OFF|DATA|MEMory|BOTH
Preset Value	DATA
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.DISPlay.USER1.TRACe8.MODE = Var
Var = SCPI.DISPlay.USER1.TRACe8.MODE
```

Related Objects

Equivalent Key

USER Menu: **Trace View** > **Display Trace**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:TRACe[1-8]:MODE {OFF|DATA|MEMory|BOTH}

:DISPlay:USER[1-1]:TRACe[1-8]:MODE?

Query Response

{OFF|DATA|MEMory|BOTH} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:TRACe8:MODE OFF"
20 OUTPUT 717;":DISPlay:USER1:TRACe8:MODE ?"
30 ENTER 717;A$
```

SCPI.DISPlay.USER(Ch).TRACe(Tr).PERSistence.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).TRACe(Tr).PERSistence.STATe = *Value*

Value = SCPI.DISPlay.USER(Ch).TRACe(Tr).PERSistence.STATe

Description

This command sets/gets trace storage mode, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace storage mode
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.USER1.TRACe8.PERSistence.STATe = Var
Var = SCPI.DISPlay.USER1.TRACe8.PERSistence.STATe
```

Related Objects

Equivalent Key

USER Menu: **Trace View** > **Persistence** > **Persistence Mode**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:TRACe[1-8]:PERSistence:STATe {ON|OFF|1|0}

:DISPlay:USER[1-1]:TRACe[1-8]:PERSistence:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:TRACe8:PERSistence:STATe 1"
20 OUTPUT 717;":DISPlay:USER1:TRACe8:PERSistence:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.USER(Ch).TRACe(Tr).STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).TRACe(Tr).STATe = *Value*

Value = SCPI.DISPlay.USER(Ch).TRACe(Tr).STATe

Description

This command sets/gets trace state i.e. enable/disable. At least one trace must be turned on., for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trace state
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	61, "Failed to hide trace"

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.USER1.TRACe8.STATe = Var
Var = SCPI.DISPlay.USER1.TRACe8.STATe
```

Equivalent Key

USER Menu: **Trace View** > **Enable Trace** > **Trace 1**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:TRACe[1-8]:STATe {ON|OFF|1|0}

:DISPlay:USER[1-1]:TRACe[1-8]:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:TRACe8:STATe 1"
20 OUTPUT 717;":DISPlay:USER1:TRACe8:STATe ?"
30 ENTER 717;A
```

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.SCALe.AUTO

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.SCALe.AUTO = *Value*

Value = SCPI.DISPlay.USER(Ch).TRACe(Tr).X.SCALe.AUTO

Description

This command sets/gets autoscale setting for X axis, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Autoscale setting for X axis
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.DISPlay.USER1.TRACe8.X.SCALe.AUTO = Var
Var = SCPI.DISPlay.USER1.TRACe8.X.SCALe.AUTO
```

Related Objects

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.SCALe.LEFT

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.SCALe.RIGHt

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.TYPE

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.UNIT

Equivalent Key

USER Menu: **Scale** > **X Axis** > **Auto**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:TRACe[1-8]:X[:SCALe]:AUTO {ON|OFF|1|0}

:DISPlay:USER[1-1]:TRACe[1-8]:X[:SCALe]:AUTO?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:TRACe8:X[:SCALe]:AUTO 1"
20 OUTPUT 717;":DISPlay:USER1:TRACe8:X[:SCALe]:AUTO ?"
30 ENTER 717;A
```

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.SCALe.LEFT

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.SCALe.LEFT = *Value*

Value = SCPI.DISPlay.USER(Ch).TRACe(Tr).X.SCALe.LEFT

Description

This command sets/gets left setting for X axis , for the selected trace *Tr* of the selected channel *Ch*.

NOTE This command is available only when the automatic setting of the X-axis display range (set with SCPI.DISPlay.USER(Ch).TRACe(Tr).X.SCALe.AUTO) is set to OFF.

Variable

Parameter	*Value*
Description	Left setting for X axis
Data Type	Double precision floating point type (Double)
Range	-500G ~ 499.999999999G
Preset Value	0
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
Var= -500000000000
SCPI.DISPlay.USER1.TRACe8.X.SCALe.LEFT = Var
Var = SCPI.DISPlay.USER1.TRACe8.X.SCALe.LEFT
```

Related Objects

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.SCALe.AUTO

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.SCALe.RIGHt

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.TYPE

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.UNIT

Equivalent Key

USER Menu: **Scale** > **X Axis** > **Left**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:TRACe[1-8]:X[:SCALe]:LEFT {-500G ~ 499.999999999G}

:DISPlay:USER[1-1]:TRACe[1-8]:X[:SCALe]:LEFT?

Query Response

{-500G ~ 499.999999999G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:TRACe8:X[:SCALe]:LEFT -500000000000"
20 OUTPUT 717;":DISPlay:USER1:TRACe8:X[:SCALe]:LEFT ?"
30 ENTER 717;A
```

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.SCALe.RIGHt

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.SCALe.RIGHt = *Value*

Value = SCPI.DISPlay.USER(Ch).TRACe(Tr).X.SCALe.RIGHt

Description

This command sets/gets right setting for X axis, for the selected trace *Tr* of the selected channel *Ch*.

NOTE This command is available only when the automatic setting of the X-axis display range (set with SCPI.DISPlay.USER(Ch).TRACe(Tr).X.SCALe.AUTO) is set to OFF.

Variable

Parameter	*Value*
Description	Right setting for X axis
Data Type	Double precision floating point type (Double)
Range	-499.999999999G ~ 500G
Preset Value	100
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
Var= -499999999999
SCPI.DISPlay.USER1.TRACe8.X.SCALe.RIGHt = Var
Var = SCPI.DISPlay.USER1.TRACe8.X.SCALe.RIGHt
```

Related Objects

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.SCALe.AUTO

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.SCALe.LEFT

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.TYPE

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.UNIT

Equivalent Key

USER Menu: **Scale** > **X Axis** > **Right**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:TRACe[1-8]:X[:SCALe]:RIGHt {-499.999999999G ~ 500G}

:DISPlay:USER[1-1]:TRACe[1-8]:X[:SCALe]:RIGHt?

Query Response

{-499.999999999G ~ 500G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:TRACe8:X[:SCALe]:RIGHt -499999999999"
20 OUTPUT 717;":DISPlay:USER1:TRACe8:X[:SCALe]:RIGHt ?"
30 ENTER 717;A
```

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.TYPE

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.TYPE = *Value*

Value = SCPI.DISPlay.USER(Ch).TRACe(Tr).X.TYPE

Description

This command sets/gets x axis type, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	X axis type
Data Type	Character string type (String)
Range	LINear|LOGarithmic
Preset Value	LINear
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "LINear"
SCPI.DISPlay.USER1.TRACe8.X.TYPE = Var
Var = SCPI.DISPlay.USER1.TRACe8.X.TYPE
```

Related Objects

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.SCALe.AUTO

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.SCALe.LEFT

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.SCALe.RIGHt

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.UNIT

Equivalent Key

USER Menu: **Scale** > **X Axis Type**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:TRACe[1-8]:X:TYPE {LINear|LOGarithmic}

:DISPlay:USER[1-1]:TRACe[1-8]:X:TYPE?

Query Response

{LINear|LOGarithmic} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:TRACe8:X:TYPE LINear"
20 OUTPUT 717;":DISPlay:USER1:TRACe8:X:TYPE ?"
30 ENTER 717;A$
```

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.UNIT

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.UNIT = *Value*

Value = SCPI.DISPlay.USER(Ch).TRACe(Tr).X.UNIT

Description

This command sets/gets X axis unit, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	X axis unit
Data Type	Character string type (String)
Range	254 chars
Preset Value	"U"
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
SCPI.DISPlay.USER1.TRACe8.X.UNIT = Var
Var = SCPI.DISPlay.USER1.TRACe8.X.UNIT
```

Related Objects

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.SCALe.AUTO

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.SCALe.LEFT

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.SCALe.RIGHt

SCPI.DISPlay.USER(Ch).TRACe(Tr).X.TYPE

Equivalent Key

USER Menu: **Scale** > **X Unit**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:TRACe[1-8]:X:UNIT

:DISPlay:USER[1-1]:TRACe[1-8]:X:UNIT?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:TRACe8:X:UNIT Sample_Text"
20 OUTPUT 717;":DISPlay:USER1:TRACe8:X:UNIT ?"
30 ENTER 717;A$
```

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.AUTO

Object Type

Method (**Write Only**)

Syntax

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.AUTO

Description

This command sets auto scale setting for Y axis, for the selected trace *Tr* of the selected channel *Ch*.

Examples

SCPI.DISPlay.USER1.TRACe8.Y.SCALe.AUTO

Related Objects

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.RLEVel

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.RPOSition

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.UNIT

Equivalent Key

USER Menu: **Scale** > **Auto Scale**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:TRACe[1-8]:Y[:SCALe]:AUTO

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:TRACe8:Y[:SCALe]:AUTO"
```

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.PDIVision

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.PDIVision = *Value*

Value = SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.PDIVision

Description

This command sets/gets scale per division, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale per division
Data Type	Double precision floating point type (Double)
Range	1a ~ 10G
Preset Value	10
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0.000000000001
SCPI.DISPlay.USER1.TRACe8.Y.SCALe.PDIVision = Var
Var = SCPI.DISPlay.USER1.TRACe8.Y.SCALe.PDIVision
```

Related Objects

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.RLEVel

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.RPOSition

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.UNIT

Equivalent Key

USER Menu: **Scale** > **Scale/Div**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:TRACe[1-8]:Y[:SCALe]:PDIVision {1a ~ 10G}

:DISPlay:USER[1-1]:TRACe[1-8]:Y[:SCALe]:PDIVision?

Query Response

{1a ~ 10G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:TRACe8:Y[:SCALe]:PDIVision 0.000000000001"
20 OUTPUT 717;":DISPlay:USER1:TRACe8:Y[:SCALe]:PDIVision ?"
30 ENTER 717;A
```

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.RLEVel

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.RLEVel = *Value*

Value = SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.RLEVel

Description

This command sets/gets scale reference level, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale reference level
Data Type	Double precision floating point type (Double)
Range	-500G ~ 500G
Preset Value	-40
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -500000000000
SCPI.DISPlay.USER1.TRACe8.Y.SCALe.RLEVel = Var
Var = SCPI.DISPlay.USER1.TRACe8.Y.SCALe.RLEVel
```

Related Objects

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.RPOSition

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.UNIT

Equivalent Key

USER Menu: **Scale** > **Reference Value**

USER Menu: **Scale** > **Marker** > **Reference**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:TRACe[1-8]:Y[:SCALe]:RLEVel {-500G ~ 500G}

:DISPlay:USER[1-1]:TRACe[1-8]:Y[:SCALe]:RLEVel?

Query Response

{-500G ~ 500G} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:TRACe8:Y[:SCALe]:RLEVel -500000000000"
20 OUTPUT 717;":DISPlay:USER1:TRACe8:Y[:SCALe]:RLEVel ?"
30 ENTER 717;A
```

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.RPOSition

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.RPOSition = *Value*

Value = SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.RPOSition

Description

This command sets/gets scale reference position, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Scale reference position
Data Type	Long integer type (Long)
Range	0 ~ 30
Preset Value	5
Unit	Div
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.DISPlay.USER1.TRACe8.Y.SCALe.RPOSition = Var
Var = SCPI.DISPlay.USER1.TRACe8.Y.SCALe.RPOSition
```

Related Objects

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.RLEVel

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.UNIT

Equivalent Key

USER Menu: **Scale** > **Reference Position**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:TRACe[1-8]:Y[:SCALe]:RPOSition {0 ~ 30}

:DISPlay:USER[1-1]:TRACe[1-8]:Y[:SCALe]:RPOSition?

Query Response

{0 ~ 30} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:TRACe8:Y[:SCALe]:RPOSition 0 "
20 OUTPUT 717;":DISPlay:USER1:TRACe8:Y[:SCALe]:RPOSition ?"
30 ENTER 717;A
```

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.UNIT

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.UNIT = *Value*

Value = SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.UNIT

Description

This command sets/gets Y axis unit, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Y axis unit
Data Type	Character string type (String)
Range	254 chars
Preset Value	"U"
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
SCPI.DISPlay.USER1.TRACe8.Y.UNIT = Var
Var = SCPI.DISPlay.USER1.TRACe8.Y.UNIT
```

Related Objects

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.AUTO

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.PDIVision

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.RLEVel

SCPI.DISPlay.USER(Ch).TRACe(Tr).Y.SCALe.RPOSition

Equivalent Key

USER Menu: **Scale** > **Y Unit**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:TRACe[1-8]:Y:UNIT

:DISPlay:USER[1-1]:TRACe[1-8]:Y:UNIT?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:TRACe8:Y:UNIT Sample_Text"
20 OUTPUT 717;":DISPlay:USER1:TRACe8:Y:UNIT ?"
30 ENTER 717;A$
```

SCPI.DISPlay.USER(Ch).Y.SCALe.DIVisions

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.USER(Ch).Y.SCALe.DIVisions = *Value*

Value = SCPI.DISPlay.USER(Ch).Y.SCALe.DIVisions

Description

This command sets/gets number of Y division, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Number of Y division
Data Type	Long integer type (Long)
Range	4 ~ 30
Preset Value	10
Unit	-
Resolution	2
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 4
SCPI.DISPlay.USER1.Y.SCALe.DIVisions = Var
Var = SCPI.DISPlay.USER1.Y.SCALe.DIVisions
```

Equivalent Key

USER Menu: **Scale** > **Divisions**

Equivalent SCPI Command

Syntax

:DISPlay:USER[1-1]:Y[:SCALe]:DIVisions {4 ~ 30}

:DISPlay:USER[1-1]:Y[:SCALe]:DIVisions?

Query Response

{4 ~ 30} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:USER1:Y[:SCALe]:DIVisions 4 "
20 OUTPUT 717;":DISPlay:USER1:Y[:SCALe]:DIVisions ?"
30 ENTER 717;A
```

SCPI.DISPlay.WINDow.ACTive

Object Type

Property (**Read-Write**)

Syntax

SCPI.DISPlay.WINDow.ACTive = *Value*

Value = SCPI.DISPlay.WINDow.ACTive

Description

This command sets/gets the specified window visible and active.

NOTE When trigger couple is on and trigger signal is entered continuously, the measurement is made immediately and the trace is overwritten after this command is executed.

Variable

Parameter	*Value*
Description	Specified window visible and active
Data Type	Character string type (String)
Range	PN1|SP1|FP1|TR1|AM1|BB1|PS1|USER1
Preset Value	PN1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	100, "Option not installed"

Examples

```
Dim Var as String
Var= "PN1"
SCPI.DISPlay.WINDow.ACTive = Var
Var = SCPI.DISPlay.WINDow.ACTive
```

Equivalent Key

PN Menu: **Measurement View** > **Phase Noise**

PN Menu: **Measurement View** > **Spectrum Monitor**

PN Menu: **Measurement View** > **Freq & Power**

PN Menu: **Measurement View** > **Transient**

PN Menu: **Measurement View** > **AM Noise**

PN Menu: **Measurement View** > **Baseband Noise**

PN Menu: **Measurement View** > **Segment PN**

PN Menu: **Measurement View** > **User**

SP Menu: **Measurement View** > **Phase Noise**

SP Menu: **Measurement View** > **Spectrum Monitor**

SP Menu: **Measurement View** > **Freq & Power**

SP Menu: **Measurement View** > **Transient**

SP Menu: **Measurement View** > **AM Noise**

SP Menu: **Measurement View** > **Baseband Noise**

SP Menu: **Measurement View** > **Segment PN**

SP Menu: **Measurement View** > **User**

FP Menu: **Measurement View** > **Phase Noise**

FP Menu: **Measurement View** > **Spectrum Monitor**

FP Menu: **Measurement View** > **Freq & Power**

FP Menu: **Measurement View** > **Transient**

FP Menu: **Measurement View** > **AM Noise**

FP Menu: **Measurement View** > **Baseband Noise**

FP Menu: **Measurement View** > **Segment PN**

FP Menu: **Measurement View** > **User**

TR Menu: **Measurement View** > **Phase Noise**

TR Menu: **Measurement View** > **Spectrum Monitor**

TR Menu: **Measurement View** > **Freq & Power**

TR Menu: **Measurement View** > **Transient**

TR Menu: **Measurement View** > **AM Noise**

TR Menu: **Measurement View** > **Baseband Noise**

TR Menu: **Measurement View** > **Segment PN**

TR Menu: **Measurement View** > **User**

AM Menu: **Measurement View** > **Phase Noise**

AM Menu: **Measurement View** > **Spectrum Monitor**

AM Menu: **Measurement View** > **Freq & Power**

AM Menu: **Measurement View** > **Transient**

AM Menu: **Measurement View** > **AM Noise**

AM Menu: **Measurement View** > **Baseband Noise**

AM Menu: **Measurement View** > **Segment PN**

AM Menu: **Measurement View** > **User**

BB Menu: **Measurement View** > **Phase Noise**

BB Menu: **Measurement View** > **Spectrum Monitor**

BB Menu: **Measurement View** > **Freq & Power**

BB Menu: **Measurement View** > **Transient**

BB Menu: **Measurement View** > **AM Noise**

BB Menu: **Measurement View** > **Baseband Noise**

BB Menu: **Measurement View** > **Segment PN**

BB Menu: **Measurement View** > **User**

PS Menu: **Measurement View** > **Phase Noise**

PS Menu: **Measurement View** > **Spectrum Monitor**

PS Menu: **Measurement View** > **Freq & Power**

PS Menu: **Measurement View** > **Transient**

PS Menu: **Measurement View** > **AM Noise**

PS Menu: **Measurement View** > **Baseband Noise**

PS Menu: **Measurement View** > **Segment PN**

PS Menu: **Measurement View** > **User**

USER Menu: **Measurement View** > **Phase Noise**

USER Menu: **Measurement View** > **Spectrum Monitor**

USER Menu: **Measurement View** > **Freq & Power**

USER Menu: **Measurement View** > **Transient**

USER Menu: **Measurement View** > **AM Noise**

USER Menu: **Measurement View** > **Baseband Noise**

USER Menu: **Measurement View** > **Segment PN**

USER Menu: **Measurement View** > **User**

Equivalent SCPI Command

Syntax

:DISPlay:WINDow:ACTive {PN1|SP1|FP1|TR1|AM1|BB1|PS1|USER1}

:DISPlay:WINDow:ACTive?

Query Response

{PN1|SP1|FP1|TR1|AM1|BB1|PS1|USER1} <newline><^END>

Example of use

```
10 OUTPUT 717;":DISPlay:WINDow:ACTive PN1"
20 OUTPUT 717;":DISPlay:WINDow:ACTive ?"
30 ENTER 717;A$
```

Format

SCPI.FORMat.BORDer

Object Type

Property (**Read-Write**)

Syntax

SCPI.FORMat.BORDer = *Value*

Value = SCPI.FORMat.BORDer

Description

This command sets/gets byte order setting for binary transfer. Applied for all SCPI connections and array commands/queries.

Variable

Parameter	*Value*
Description	Byte order setting for binary transfer
Data Type	Character string type (String)
Range	NORMal: Set byte order so that a byte containing MSB (Most Significant Bit) is transferred first SWAPped: Set byte order so that a byte containing LSB (Least Significant Bit) is transferred first
Preset Value	NORMal
Unit	-
Resolution	-
Preset Effect	Yes

Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NORMal"
SCPI.FORMat.BORDer = Var
Var = SCPI.FORMat.BORDer
```

Related Objects

SCPI.FORMat.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:FORMat:BORDer {NORMal|SWAPped}

:FORMat:BORDer?

Query Response

{NORMal|SWAPped} <newline><^END>

Example of use

```
10 OUTPUT 717;":FORMat:BORDer NORMal"
20 OUTPUT 717;":FORMat:BORDer ?"
30 ENTER 717;A$
```

SCPI.FORMat.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.FORMat.DATA = *Value*

Value = SCPI.FORMat.DATA

Description

This command sets/gets a data transfer mode applied for all SCPI connections, applied for all array commands/queries.

Variable

Parameter	*Value*
Description	Data transfer mode
Data Type	Character string type (String)
Range	ASCii: Set data transfer mode to 'ASCii' REAL32:Set data transfer mode to 'IEEE 32 floating point binay' REAL64:Set data transfer mode to 'IEEE 64 floating point binay'
Preset Value	ASCii
Unit	-
Resolution	-
Preset Effect	Yes

Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "ASCii"
SCPI.FORMat.DATA = Var
Var = SCPI.FORMat.DATA
```

Related Objects

SCPI.FORMat.BORDer

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:FORMat:DATA {ASCii|REAL32|REAL64}

:FORMat:DATA?

Query Response

{ASCii|REAL32|REAL64} <newline><^END>

Example of use

```
10 OUTPUT 717;":FORMat:DATA ASCii"
20 OUTPUT 717;":FORMat:DATA ?"
30 ENTER 717;A$
```

Hcopy

SCPI.HCOPy.ABORt

Object Type

Method (**Write Only**)

Syntax

SCPI.HCOPy.ABORt

Description

This command aborts printing.

Examples

SCPI.HCOPy.ABORt

Related Objects

SCPI.HCOPy.IMMediate

Equivalent Key

PN Menu: **System** > **Abort Printing**

SP Menu: **System** > **Abort Printing**

FP Menu: **System** > **Abort Printing**

TR Menu: **System** > **Abort Printing**

AM Menu: **System** > **Abort Printing**

BB Menu: **System** > **Abort Printing**

PS Menu: **System** > **Abort Printing**

USER Menu: **System** > **Abort Printing**

Equivalent SCPI Command

Syntax

:HCOPy:ABORt

Example of use

```
10 OUTPUT 717;":HCOPy:ABORt"
```

SCPI.HCOPy.IMAGe

Object Type

Property (**Read-Write**)

Syntax

SCPI.HCOPy.IMAGe = *Value*

Value = SCPI.HCOPy.IMAGe

Description

This command sets/gets print color for output (to the printer).

Variable

Parameter	*Value*
Description	Print color for output to the printer
Data Type	Character string type (String)
Range	NORMal|INVert
Preset Value	NORMal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NORMal"
SCPI.HCOPy.IMAGe = Var
Var = SCPI.HCOPy.IMAGe
```

Related Objects

SCPI.HCOPy.IMMediate

Equivalent Key

PN Menu: **System** > **Invert Image**

SP Menu: **System** > **Invert Image**

FP Menu: **System** > **Invert Image**

TR Menu: **System** > **Invert Image**

AM Menu: **System** > **Invert Image**

BB Menu: **System** > **Invert Image**

PS Menu: **System** > **Invert Image**

USER Menu: **System** > **Invert Image**

Equivalent SCPI Command

Syntax

:HCOPy:IMAGe {NORMal|INVert}

:HCOPy:IMAGe?

Query Response

{NORMal|INVert} <newline><^END>

Example of use

```
10 OUTPUT 717;":HCOPy:IMAGe NORMal"
20 OUTPUT 717;":HCOPy:IMAGe ?"
30 ENTER 717;A$
```

SCPI.HCOPy.IMMediate

Object Type

Method (**Write Only**)

Syntax

SCPI.HCOPy.IMMediate

Description

This command sets the display image on the LCD display to the printer connected to the E5052B.

Examples

SCPI.HCOPy.IMMediate

Related Objects

SCPI.HCOPy.ABORt

SCPI.HCOPy.IMAGe

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:HCOPy:IMMediate

Example of use

```
10 OUTPUT 717;":HCOPy:IMMediate"
```

SCPI.HCOPy.SDUMp.DATA (fmt)

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.HCOPy.SDUMp.DATA fmt

Description

This command gets query display image data.

Variable

Parameter	*fmt*
Description	display image data
Data Type	Character string type (String)
Range	GIF|BMP|PNG
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-200, "Execution error"

Examples

```
Dim Var as String
Var= "PNG"
Var = SCPI.HCOPy.SDUMp.DATA_Q fmt, data
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:HCOPy:SDUMp:DATA?

Query Response

{GIF|BMP|PNG -} <newline><^END>

Example of use

```
10 OUTPUT 717;":HCOPy:SDUMp:DATA ?"
```

Initiate

SCPI.INITiate.AM(Ch).CONTinuous

Object Type

Property (**Read-Write**)

Syntax

SCPI.INITiate.AM(Ch).CONTinuous = *Value*

Value = SCPI.INITiate.AM(Ch).CONTinuous

Description

This command sets/gets a property to always move to waiting-for-trigger state after measuring, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Waiting-for-trigger state after measurement
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.INITiate.AM1.CONTinuous = Var
Var = SCPI.INITiate.AM1.CONTinuous
```

Related Objects

SCPI.INITiate.AM(Ch).IMMediate

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:INITiate:AM[1-1]:CONTinuous {ON|OFF|1|0}

:INITiate:AM[1-1]:CONTinuous?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":INITiate:AM1:CONTinuous 1"
20 OUTPUT 717;":INITiate:AM1:CONTinuous ?"
30 ENTER 717;A
```

SCPI.INITiate.AM(Ch).IMMediate

Object Type

Method (**Write Only**)

Syntax

SCPI.INITiate.AM(Ch).IMMediate

Description

This command sets the property to move once to waiting-for-trigger state, for the selected channel *Ch*.

Examples

SCPI.INITiate.AM1.IMMediate

Related Objects

SCPI.INITiate.AM(Ch).CONTinuous

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:INITiate:AM[1-1]:IMMediate

Example of use

```
10 OUTPUT 717;":INITiate:AM1:IMMediate"
```

SCPI.INITiate.BB(Ch).CONTinuous

Object Type

Property (**Read-Write**)

Syntax

SCPI.INITiate.BB(Ch).CONTinuous = *Value*

Value = SCPI.INITiate.BB(Ch).CONTinuous

Description

This command sets/gets always move to waiting-for-trigger state after measuring, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Waiting-for-trigger state after measurement
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.INITiate.BB1.CONTinuous = Var
Var = SCPI.INITiate.BB1.CONTinuous
```

Related Objects

SCPI.INITiate.BB(Ch).IMMediate

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:INITiate:BB[1-1]:CONTinuous {ON|OFF|1|0}

:INITiate:BB[1-1]:CONTinuous?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":INITiate:BB1:CONTinuous 1"
20 OUTPUT 717;":INITiate:BB1:CONTinuous ?"
30 ENTER 717;A
```

SCPI.INITiate.BB(Ch).IMMediate

Object Type

Method (**Write Only**)

Syntax

SCPI.INITiate.BB(Ch).IMMediate

Description

This command sets the property to move once to waiting-for-trigger state, for the selected channel *Ch*.

Examples

SCPI.INITiate.BB1.IMMediate

Related Objects

SCPI.INITiate.BB(Ch).CONTinuous

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:INITiate:BB[1-1]:IMMediate

Example of use

```
10 OUTPUT 717;":INITiate:BB1:IMMediate"
```

SCPI.INITiate.FP(Ch).CONTinuous

Object Type

Property (**Read-Write**)

Syntax

SCPI.INITiate.FP(Ch).CONTinuous = *Value*

Value = SCPI.INITiate.FP(Ch).CONTinuous

Description

This command sets/gets always move to waiting-for-trigger state after measuring, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Waiting-for-trigger state after measurement
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.INITiate.FP1.CONTinuous = Var
Var = SCPI.INITiate.FP1.CONTinuous
```

Related Objects

SCPI.INITiate.FP(Ch).IMMediate

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:INITiate:FP[1-1]:CONTinuous {ON|OFF|1|0}

:INITiate:FP[1-1]:CONTinuous?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":INITiate:FP1:CONTinuous 1"
20 OUTPUT 717;":INITiate:FP1:CONTinuous ?"
30 ENTER 717;A
```

SCPI.INITiate.FP(Ch).IMMediate

Object Type

Method (**Write Only**)

Syntax

SCPI.INITiate.FP(Ch).IMMediate

Description

This command sets the property to move once to waiting-for-trigger state, for the selected channel *Ch*.

Examples

SCPI.INITiate.FP1.IMMediate

Related Objects

SCPI.INITiate.FP(Ch).CONTinuous

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:INITiate:FP[1-1]:IMMediate

Example of use

```
10 OUTPUT 717;":INITiate:FP1:IMMediate"
```

SCPI.INITiate.PN(Ch).CONTinuous

Object Type

Property (**Read-Write**)

Syntax

SCPI.INITiate.PN(Ch).CONTinuous = *Value*

Value = SCPI.INITiate.PN(Ch).CONTinuous

Description

This command sets/gets always move to waiting-for-trigger state after measuring, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Waiting-for-trigger state after measurement
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.INITiate.PN1.CONTinuous = Var
Var = SCPI.INITiate.PN1.CONTinuous
```

Related Objects

SCPI.INITiate.PN(Ch).IMMediate

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:INITiate:PN[1-1]:CONTinuous {ON|OFF|1|0}

:INITiate:PN[1-1]:CONTinuous?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":INITiate:PN1:CONTinuous 1"
20 OUTPUT 717;":INITiate:PN1:CONTinuous ?"
30 ENTER 717;A
```

SCPI.INITiate.PN(Ch).IMMediate

Object Type

Method (**Write Only**)

Syntax

SCPI.INITiate.PN(Ch).IMMediate

Description

This command sets the property to move once to waiting-for-trigger state, for the selected channel *Ch*.

Examples

SCPI.INITiate.PN1.IMMediate

Related Objects

SCPI.INITiate.PN(Ch).CONTinuous

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:INITiate:PN[1-1]:IMMediate

Example of use

```
10 OUTPUT 717;":INITiate:PN1:IMMediate"
```

SCPI.INITiate.PS(Ch).CONTinuous

Object Type

Property (**Read-Write**)

Syntax

SCPI.INITiate.PS(Ch).CONTinuous = *Value*

Value = SCPI.INITiate.PS(Ch).CONTinuous

Description

This command sets/gets always move to waiting-for-trigger state after measuring, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Waiting-for-trigger state after measurement
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.INITiate.PS1.CONTinuous = Var
Var = SCPI.INITiate.PS1.CONTinuous
```

Related Objects

SCPI.INITiate.PS(Ch).IMMediate

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:INITiate:PS[1-1]:CONTinuous {ON|OFF|1|0}

:INITiate:PS[1-1]:CONTinuous?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":INITiate:PS1:CONTinuous 1"
20 OUTPUT 717;":INITiate:PS1:CONTinuous?"
30 ENTER 717;A
```

SCPI.INITiate.PS(Ch).IMMediate

Object Type

Method (**Write Only**)

Syntax

SCPI.INITiate.PS(Ch).IMMediate

Description

This command sets the property to move once to waiting-for-trigger state, for the selected channel *Ch*.

Examples

SCPI.INITiate.PS1.IMMediate

Related Objects

SCPI.INITiate.PS(Ch).CONTinuous

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:INITiate:PS[1-1]:IMMediate

Example of use

```
10 OUTPUT 717;":INITiate:PS1:IMMediate"
```

SCPI.INITiate.SP(Ch).CONTinuous

Object Type

Property (**Read-Write**)

Syntax

SCPI.INITiate.SP(Ch).CONTinuous = *Value*

Value = SCPI.INITiate.SP(Ch).CONTinuous

Description

This command sets/gets always move to waiting-for-trigger state after measuring, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Waiting-for-trigger state after measurement
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.INITiate.SP1.CONTinuous = Var
Var = SCPI.INITiate.SP1.CONTinuous
```

Related Objects

SCPI.INITiate.SP(Ch).IMMediate

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:INITiate:SP[1-1]:CONTinuous {ON|OFF|1|0}

:INITiate:SP[1-1]:CONTinuous?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":INITiate:SP1:CONTinuous 1"
20 OUTPUT 717;":INITiate:SP1:CONTinuous ?"
30 ENTER 717;A
```

SCPI.INITiate.SP(Ch).IMMediate

Object Type

Method (**Write Only**)

Syntax

SCPI.INITiate.SP(Ch).IMMediate

Description

This command sets the property to move once to waiting-for-trigger state, for the selected channel *Ch*.

Examples

SCPI.INITiate.SP1.IMMediate

Related Objects

SCPI.INITiate.SP(Ch).CONTinuous

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:INITiate:SP[1-1]:IMMediate

Example of use

```
10 OUTPUT 717;":INITiate:SP1:IMMediate"
```

SCPI.INITiate.TR(Ch).CONTinuous

Object Type

Property (**Read-Write**)

Syntax

SCPI.INITiate.TR(Ch).CONTinuous = *Value*

Value = SCPI.INITiate.TR(Ch).CONTinuous

Description

This command sets/gets always move to waiting-for-trigger state after measuring, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Waiting-for-trigger state after measurement
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.INITiate.TR1.CONTinuous = Var
Var = SCPI.INITiate.TR1.CONTinuous
```

Related Objects

SCPI.INITiate.TR(Ch).IMMediate

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:INITiate:TR[1-1]:CONTinuous {ON|OFF|1|0}

:INITiate:TR[1-1]:CONTinuous?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":INITiate:TR1:CONTinuous 1"
20 OUTPUT 717;":INITiate:TR1:CONTinuous ?"
30 ENTER 717;A
```

SCPI.INITiate.TR(Ch).IMMediate

Object Type

Method (**Write Only**)

Syntax

SCPI.INITiate.TR(Ch).IMMediate

Description

This command sets the property to move once to waiting-for-trigger state, for the selected channel *Ch*.

NOTE

Trigger is not detected while sampling pre-trigger data. Make sure to add a certain time after initiation of trigger system when programming so that the instrument is triggered after a time for pre-trigger passed. This certain time is required when trigger source is selected, "external", "Wide Video" or "Narrow Video".

Examples

SCPI.INITiate.TR1.IMMediate

Related Objects

SCPI.INITiate.TR(Ch).CONTinuous

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:INITiate:TR[1-1]:IMMediate

Example of use

```
10 OUTPUT 717;":INITiate:TR1:IMMediate"
```

Mmemory

SCPI.MMEMory.AM(Ch).TRACe(Tr).LOAD.DATA

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.AM(Ch).TRACe(Tr).LOAD.DATA = *Value*

Description

This command recalls data trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Recalls data trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No

Error	-256, "File name not found" -257, "File name error" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.AM1.TRACe1.LOAD.DATA = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:AM[1-1]:TRACe[1-1]:LOAD:DATA

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:AM1:TRACe1:LOAD:DATA Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.AM(Ch).TRACe(Tr).LOAD.LIMit.LOWer

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.AM(Ch).TRACe(Tr).LOAD.LIMit.LOWer = *Value*

Description

This command imports lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Imports lower limit line
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.AM1.TRACe1.LOAD.LIMit.LOWer = Var
```

Related Objects

SCPI.MMEMory.AM(Ch).TRACe(Tr).LOAD.LIMit.UPPer

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:AM[1-1]:TRACe[1-1]:LOAD:LIMit:LOWer

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:AM1:TRACe1:LOAD:LIMit:LOWer Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.AM(Ch).TRACe(Tr).LOAD.LIMit.UPPer

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.AM(Ch).TRACe(Tr).LOAD.LIMit.UPPer = *Value*

Description

This command imports upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Imports upper limit line
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.AM1.TRACe1.LOAD.LIMit.UPPer = Var
```

Related Objects

SCPI.MMEMory.AM(Ch).TRACe(Tr).LOAD.LIMit.LOWer

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:AM[1-1]:TRACe[1-1]:LOAD:LIMit:UPPer

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:AM1:TRACe1:LOAD:LIMit:UPPer Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.AM(Ch).TRACe(Tr).LOAD.MEMory

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.AM(Ch).TRACe(Tr).LOAD.MEMory = *Value*

Description

This command recalls memory trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Recalls memory trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-256, "File name not found" -257, "File name error" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.AM1.TRACe1.LOAD.MEMory = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:AM[1-1]:TRACe[1-1]:LOAD:MEMory

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:AM1:TRACe1:LOAD:MEMory Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.AM(Ch).TRACe(Tr).LOAD.SPURious.OSSPur.FLISt

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.AM(Ch).TRACe(Tr).LOAD.SPURious.OSSPur.FLISt = *Value*

Description

This command Imports Spurious Table, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Imports Spurious Table
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-256, "File name not found" -257, "File name error" 70,

	"Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.AM1.TRACe1.LOAD.SPURious.OSSPur.FLISt = Var
```

Related Objects

SCPI.MMEMory.AM(Ch).TRACe(Tr).LOAD.SPURious.THReshold

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:AM[1-1]:TRACe[1-1]:LOAD:SPURious:OSSPur:FLISt

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:AM1:TRACe1:LOAD:SPURious:OSSPur:FLISt Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.AM(Ch).TRACe(Tr).LOAD.SPURious.THReshold

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.AM(Ch).TRACe(Tr).LOAD.SPURious.THReshold = *Value*

Description

This command imports threshold table, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Imports threshold table
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 70,

	"Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.AM1.TRACe1.LOAD.SPURious.THReshold = Var
```

Related Objects

SCPI.MMEMory.AM(Ch).TRACe(Tr).LOAD.SPURious.OSSPur.FLISt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:AM[1-1]:TRACe[1-1]:LOAD:SPURious:THReshold

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:AM1:TRACe1:LOAD:SPURious:THReshold Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.AM(Ch).TRACe(Tr).STORe.DATA

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.AM(Ch).TRACe(Tr).STORe.DATA = *Value*

Description

This command saves active trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Saves active trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 71, "Failed to write file" 76, "Save failed"

Examples

```
Dim Var as String
SCPI.MMEMory.AM1.TRACe1.STORe.DATA = Var
```

Related Objects

SCPI.MMEMory.AM(Ch).TRACe(Tr).STORe.DATA

SCPI.MMEMory.AM(Ch).TRACe(Tr).STORe.MEMory

SCPI.MMEMory.AM(Ch).TRACe(Tr).STORe.SPURious.OSSPur.FLISt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:AM[1-1]:TRACe[1-1]:STORe[:DATA]

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:AM1:TRACe1:STORe[:DATA] Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.AM(Ch).TRACe(Tr).STORe.MEMory

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.AM(Ch).TRACe(Tr).STORe.MEMory = *Value*

Description

This command saves active trace memory, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Saves active trace memory
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 71, "Failed to write file" 76, "Save failed"

Examples

```
Dim Var as String
SCPI.MMEMory.AM1.TRACe1.STORe.MEMory = Var
```

Related Objects

SCPI.MMEMory.AM(Ch).TRACe(Tr).STORe.DATA

SCPI.MMEMory.AM(Ch).TRACe(Tr).STORe.SPURious.OSSPur.FLISt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:AM[1-1]:TRACe[1-1]:STORe:MEMory

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:AM1:TRACe1:STORe:MEMory Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.AM(Ch).TRACe(Tr).STORe.SPURious.OSSPur.FLISt

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.AM(Ch).TRACe(Tr).STORe.SPURious.OSSPur.FLISt = *Value*

Description

This command exports Spurious Table, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Exports Spurious Table
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 71, "Failed to write file"

Examples

```
Dim Var as String
SCPI.MMEMory.AM1.TRACe1.STORe.SPURious.OSSPur.FLISt = Var
```

Related Objects

SCPI.MMEMory.AM(Ch).TRACe(Tr).STORe.DATA

SCPI.MMEMory.AM(Ch).TRACe(Tr).STORe.MEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:AM[1-1]:TRACe[1-1]:STORe:SPURious:OSSPur:FLISt

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:AM1:TRACe1:STORe:SPURious:OSSPur:FLISt Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.DATA

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.DATA = *Value*

Description

This command Recalls data trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Recalls data trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-256, "File name not found" -257, "File name error" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.BB1.TRACe1.LOAD.DATA = Var
```

Related Objects

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.DATA

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.LIMit.LOWer

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.LIMit.UPPer

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.MEMory

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.SPURious.OSSPur.FLISt

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.SPURious.THReshold

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:BB[1-1]:TRACe[1-1]:LOAD:DATA

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:BB1:TRACe1:LOAD:DATA Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.LIMit.LOWer

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.LIMit.LOWer = *Value*

Description

This command imports lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Imports lower limit line
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.BB1.TRACe1.LOAD.LIMit.LOWer = Var
```

Related Objects

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.DATA

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.LIMit.UPPer

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.MEMory

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.SPURious.OSSPur.FLISt

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.SPURious.THReshold

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:BB[1-1]:TRACe[1-1]:LOAD:LIMit:LOWer

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:BB1:TRACe1:LOAD:LIMit:LOWer Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.LIMit.UPPer

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.LIMit.UPPer = *Value*

Description

This command imports upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Imports upper limit line
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.BB1.TRACe1.LOAD.LIMit.UPPer = Var
```

Related Objects

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.DATA

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.LIMit.LOWer

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.MEMory

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.SPURious.OSSPur.FLISt

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.SPURious.THReshold

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:BB[1-1]:TRACe[1-1]:LOAD:LIMit:UPPer

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:BB1:TRACe1:LOAD:LIMit:UPPer Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.MEMory

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.MEMory = *Value*

Description

This command recalls memory trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Recall memory trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-256, "File name not found" -257, "File name error" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.BB1.TRACe1.LOAD.MEMory = Var
```

Related Objects

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.DATA

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.LIMit.LOWer

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.LIMit.UPPer

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.SPURious.OSSPur.FLISt

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.SPURious.THReshold

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:BB[1-1]:TRACe[1-1]:LOAD:MEMory

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:BB1:TRACe1:LOAD:MEMory Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.SPURious.OSSPur.FLISt

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.SPURious.OSSPur.FLISt = *Value*

Description

This command imports Spurious Table, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Imports Spurious Table
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-256, "File name not found" -257, "File name error" 70,

	"Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.BB1.TRACe1.LOAD.SPURious.OSSPur.FLISt = Var
```

Related Objects

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.DATA

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.LIMit.LOWer

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.LIMit.UPPer

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.MEMory

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.SPURious.THReshold

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:BB[1-1]:TRACe[1-1]:LOAD:SPURious:OSSPur:FLISt

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:BB1:TRACe1:LOAD:SPURious:OSSPur:FLISt Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.SPURious.THReshold

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.SPURious.THReshold = *Value*

Description

This command imports threshold table, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Imports threshold table
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 70,

	"Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.BB1.TRACe1.LOAD.SPURious.THReshold = Var
```

Related Objects

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.DATA

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.LIMit.LOWer

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.LIMit.UPPer

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.MEMory

SCPI.MMEMory.BB(Ch).TRACe(Tr).LOAD.SPURious.OSSPur.FLISt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:BB[1-1]:TRACe[1-1]:LOAD:SPURious:THReshold

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:BB1:TRACe1:LOAD:SPURious:THReshold Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.BB(Ch).TRACe(Tr).STORe.DATA

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.BB(Ch).TRACe(Tr).STORe.DATA = *Value*

Description

This command saves active trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Saves active trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 71, "Failed to write file" 76, "Save failed"

Examples

```
Dim Var as String
SCPI.MMEMory.BB1.TRACe1.STORe.DATA = Var
```

Related Objects

SCPI.MMEMory.BB(Ch).TRACe(Tr).STORe.MEMory

SCPI.MMEMory.BB(Ch).TRACe(Tr).STORe.SPURious.OSSPur.FLISt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:BB[1-1]:TRACe[1-1]:STORe[:DATA]

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:BB1:TRACe1:STORe[:DATA] Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.BB(Ch).TRACe(Tr).STORe.MEMory

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.BB(Ch).TRACe(Tr).STORe.MEMory = *Value*

Description

This command saves active trace memory, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Saves active trace memory
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 71, "Failed to write file" 76, "Save failed"

Examples

```
Dim Var as String
SCPI.MMEMory.BB1.TRACe1.STORe.MEMory = Var
```

Related Objects

SCPI.MMEMory.BB(Ch).TRACe(Tr).STORe.DATA

SCPI.MMEMory.BB(Ch).TRACe(Tr).STORe.SPURious.OSSPur.FLISt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:BB[1-1]:TRACe[1-1]:STORe:MEMory

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:BB1:TRACe1:STORe:MEMory Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.BB(Ch).TRACe(Tr).STORe.SPURious.OSSPur.FLISt

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.BB(Ch).TRACe(Tr).STORe.SPURious.OSSPur.FLISt = *Value*

Description

This command exports Spurious Table, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Exports Spurious Table
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 71, "Failed to write file"

Examples

```
Dim Var as String
SCPI.MMEMory.BB1.TRACe1.STORe.SPURious.OSSPur.FLISt = Var
```

Related Objects

SCPI.MMEMory.BB(Ch).TRACe(Tr).STORe.DATA

SCPI.MMEMory.BB(Ch).TRACe(Tr).STORe.MEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:BB[1-1]:TRACe[1-1]:STORe:SPURious:OSSPur:FLISt

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:BB1:TRACe1:STORe:SPURious:OSSPur:FLISt Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.CATalog_Q dir, list

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.MMEMory.CATalog_Q dir, list

Description

This command gets Catalog directory. The return data will be in the format: ,{,}. Each indicates the name and size in bytes of one file in the directory list in the form: ;,, Directories are indicated by following back slash..

Variable

Parameter	*dir*
Description	Catalog directory
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error"

Parameter	*list*
Description	Catalog directory
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error"

Examples

```
Dim Var as Variant
Var = SCPI.MMEMory.CATalog_Q dir, list
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:CATalog?

Query Response

{254 chars 254 chars} <newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:CATalog ?"
20 ENTER 717;A$
```

SCPI.MMEMory.COPY src, dst

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.COPY src, dst = *Value*

Description

This command copies file.

Variable

Parameter	*src*
Description	Copy file
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 72 "Failed to copy file"

Parameter	*dst*
Description	Copy file
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 72 "Failed to copy file"

Examples

```
Dim Var as Variant
SCPI.MMEMory.COPY src, dst = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:COPY {254 chars 254 chars}

Query Response

{254 chars 254 chars} <newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:COPY Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.DATA[_Q] file

Object Type

Property (**Read-Write**)

Syntax

SCPI.MMEMory.DATA[_Q] file = *Value*

Value = SCPI.MMEMory.DATA[_Q] file

Description

This command sets/gets file transfer through SCPI.

NOTE The data transfer format when this commands is executed depends on the setting with the SCPI.FORMat.DATA command.

Variable

Parameter	*file*
Description	File transfer
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error"

	70, "Failed to read file" 71, "Failed to write file"

Examples

```
Dim Var as Multiple
Var= Sample_Text
SCPI.MMEMory.DATA[_Q] file, data = Var
Var = SCPI.MMEMory.DATA[_Q] file, data
```

Related Objects

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:DATA {254 chars -}

:MMEMory:DATA?

Query Response

{254 chars -} <newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:DATA Sample_Text"
20 OUTPUT 717;":MMEMory:DATA ?"
20 ENTER 717;A$
```

SCPI.MMEMory.DELete

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.DELete = *Value*

Description

This command deletes file/directory.

Variable

Parameter	*Value*
Description	Deletes file/directory
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 73, "Failed to delete file"

Examples

```
Dim Var as String
SCPI.MMEMory.DELete = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:DELete

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:DELete Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.FP(Ch).TRACe(Tr).LOAD.DATA

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.FP(Ch).TRACe(Tr).LOAD.DATA = *Value*

Description

This command recalls data trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Recalls data trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-256, "File name not found" -257, "File name error" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.FP1.TRACe4.LOAD.DATA = Var
```

Related Objects

SCPI.MMEMory.FP(Ch).TRACe(Tr).LOAD.DATA

SCPI.MMEMory.FP(Ch).TRACe(Tr).LOAD.LIMit.LOWer

SCPI.MMEMory.FP(Ch).TRACe(Tr).LOAD.LIMit.UPPer

SCPI.MMEMory.FP(Ch).TRACe(Tr).LOAD.MEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:FP[1-1]:TRACe[1-4]:LOAD:DATA

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:FP1:TRACe4:LOAD:DATA Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.FP(Ch).TRACe(Tr).LOAD.LIMit.LOWer

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.FP(Ch).TRACe(Tr).LOAD.LIMit.LOWer = *Value*

Description

This command imports lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Imports lower limit line
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.FP1.TRACe4.LOAD.LIMit.LOWer = Var
```

Related Objects

SCPI.MMEMory.FP(Ch).TRACe(Tr).LOAD.DATA

SCPI.MMEMory.FP(Ch).TRACe(Tr).LOAD.LIMit.UPPer

SCPI.MMEMory.FP(Ch).TRACe(Tr).LOAD.MEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:FP[1-1]:TRACe[1-4]:LOAD:LIMit:LOWer

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:FP1:TRACe4:LOAD:LIMit:LOWer Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.FP(Ch).TRACe(Tr).LOAD.LIMit.UPPer

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.FP(Ch).TRACe(Tr).LOAD.LIMit.UPPer = *Value*

Description

This command imports upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Imports upper limit line
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.FP1.TRACe4.LOAD.LIMit.UPPer = Var
```

Related Objects

SCPI.MMEMory.FP(Ch).TRACe(Tr).LOAD.DATA

SCPI.MMEMory.FP(Ch).TRACe(Tr).LOAD.LIMit.LOWer

SCPI.MMEMory.FP(Ch).TRACe(Tr).LOAD.MEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:FP[1-1]:TRACe[1-4]:LOAD:LIMit:UPPer

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:FP1:TRACe4:LOAD:LIMit:UPPer Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.FP(Ch).TRACe(Tr).LOAD.MEMory

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.FP(Ch).TRACe(Tr).LOAD.MEMory = *Value*

Description

This command recalls memory trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Recalls memory trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-256, "File name not found" -257, "File name error" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.FP1.TRACe4.LOAD.MEMory = Var
```

Related Objects

SCPI.MMEMory.FP(Ch).TRACe(Tr).LOAD.DATA

SCPI.MMEMory.FP(Ch).TRACe(Tr).LOAD.LIMit.LOWer

SCPI.MMEMory.FP(Ch).TRACe(Tr).LOAD.LIMit.UPPer

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:FP[1-1]:TRACe[1-4]:LOAD:MEMory

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:FP1:TRACe4:LOAD:MEMory Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.FP(Ch).TRACe(Tr).STORe.DATA

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.FP(Ch).TRACe(Tr).STORe.DATA = *Value*

Description

This command saves active trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Saves active trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 71, "Failed to write file" 76, "Save failed"

Examples

```
Dim Var as String
SCPI.MMEMory.FP1.TRACe4.STORe.DATA = Var
```

Related Objects

SCPI.MMEMory.FP(Ch).TRACe(Tr).STORe.MEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:FP[1-1]:TRACe[1-4]:STORe[:DATA]

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:FP1:TRACe4:STORe[:DATA] Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.FP(Ch).TRACe(Tr).STORe.MEMory

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.FP(Ch).TRACe(Tr).STORe.MEMory = *Value*

Description

This command saves active trace memory, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Saves active trace memory
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 71, "Failed to write file" 76, "Save failed"

Examples

```
Dim Var as String
SCPI.MMEMory.FP1.TRACe4.STORe.MEMory = Var
```

Related Objects

SCPI.MMEMory.FP(Ch).TRACe(Tr).STORe.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:FP[1-1]:TRACe[1-4]:STORe:MEMory

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:FP1:TRACe4:STORe:MEMory Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.LOAD.CORRection.POWer

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.LOAD.CORRection.POWer = *Value*

Description

This command imports power correction table.

Variable

Parameter	*Value*
Description	Imports power correction table
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.LOAD.CORRection.POWer = Var
```

Related Objects

SCPI.MMEMory.LOAD.CORRection.POWer

SCPI.MMEMory.LOAD.PROGram

SCPI.MMEMory.LOAD.STATe

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:LOAD:CORRection:POWer

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:LOAD:CORRection:POWer Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.LOAD.PROGram

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.LOAD.PROGram = *Value*

Description

This command load programs.

Variable

Parameter	*Value*
Description	Load programs
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 70, "Failed to read file" 77, "Load VBA program failed" -284, "Program currently running"

Examples

```
Dim Var as String
SCPI.MMEMory.LOAD.PROGram = Var
```

Related Objects

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:LOAD:PROGram

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:LOAD:PROGram Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.LOAD.STATe

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.LOAD.STATe = *Value*

Description

This command recalls settings.

Variable

Parameter	*Value*
Description	Recalls settings
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 75, "Recall failed"

Examples

```
Dim Var as String
SCPI.MMEMory.LOAD.STATe = Var
```

Related Objects

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:LOAD:STATe

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:LOAD:STATe Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.MDIRectory

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.MDIRectory = *Value*

Description

This command creates directory.

Variable

Parameter	*Value*
Description	Creates directory
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 74, "Failed to create directory"

Examples

```
Dim Var as String
SCPI.MMEMory.MDIRectory = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:MDIRectory

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:MDIRectory Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.PN(Ch).TRACe(Tr).LOAD.DATA

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.PN(Ch).TRACe(Tr).LOAD.DATA = *Value*

Description

This command recalls data trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Recalls data trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-256, "File name not found" -257, "File name error" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.PN1.TRACe1.LOAD.DATA = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:PN[1-1]:TRACe[1-1]:LOAD:DATA

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:PN1:TRACe1:LOAD:DATA Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.PN(Ch).TRACe(Tr).LOAD.LIMit.LOWer

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.PN(Ch).TRACe(Tr).LOAD.LIMit.LOWer = *Value*

Description

This command imports lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Lower limit line
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.PN1.TRACe1.LOAD.LIMit.LOWer = Var
```

Related Objects

SCPI.MMEMory.PN(Ch).TRACe(Tr).LOAD.LIMit.UPPer

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:PN[1-1]:TRACe[1-1]:LOAD:LIMit:LOWer

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:PN1:TRACe1:LOAD:LIMit:LOWer Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.PN(Ch).TRACe(Tr).LOAD.LIMit.UPPer

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.PN(Ch).TRACe(Tr).LOAD.LIMit.UPPer = *Value*

Description

This command imports upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Imports upper limit line
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.PN1.TRACe1.LOAD.LIMit.UPPer = Var
```

Related Objects

SCPI.MMEMory.PN(Ch).TRACe(Tr).LOAD.LIMit.LOWer

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:PN[1-1]:TRACe[1-1]:LOAD:LIMit:UPPer

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:PN1:TRACe1:LOAD:LIMit:UPPer Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.PN(Ch).TRACe(Tr).LOAD.MEMory

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.PN(Ch).TRACe(Tr).LOAD.MEMory = *Value*

Description

This command recalls memory trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Recalls memory trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-256, "File name not found" -257, "File name error" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.PN1.TRACe1.LOAD.MEMory = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:PN[1-1]:TRACe[1-1]:LOAD:MEMory

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:PN1:TRACe1:LOAD:MEMory Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.PN(Ch).TRACe(Tr).LOAD.SPURious.OSSPur.FLISt

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.PN(Ch).TRACe(Tr).LOAD.SPURious.OSSPur.FLISt = *Value*

Description

This command imports Spurious Table, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Imports Spurious Table
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-256, "File name not found" -257, "File name error" 70,

	"Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.PN1.TRACe1.LOAD.SPURious.OSSPur.FLISt = Var
```

Related Objects

SCPI.MMEMory.PN(Ch).TRACe(Tr).LOAD.SPURious.THReshold

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:PN[1-1]:TRACe[1-1]:LOAD:SPURious:OSSPur:FLISt

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:PN1:TRACe1:LOAD:SPURious:OSSPur:FLISt Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.PN(Ch).TRACe(Tr).LOAD.SPURious.THReshold

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.PN(Ch).TRACe(Tr).LOAD.SPURious.THReshold = *Value*

Description

This command imports threshold table, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Imports threshold table
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 70,

	"Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.PN1.TRACe1.LOAD.SPURious.THReshold = Var
```

Related Objects

SCPI.MMEMory.PN(Ch).TRACe(Tr).LOAD.SPURious.OSSPur.FLISt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:PN[1-1]:TRACe[1-1]:LOAD:SPURious:THReshold

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:PN1:TRACe1:LOAD:SPURious:THReshold Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.PN(Ch).TRACe(Tr).STORe.DATA

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.PN(Ch).TRACe(Tr).STORe.DATA = *Value*

Description

This command saves active trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Saves active trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 71, "Failed to write file" 76, "Save failed"

Examples

```
Dim Var as String
SCPI.MMEMory.PN1.TRACe1.STORe.DATA = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:PN[1-1]:TRACe[1-1]:STORe[:DATA]

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:PN1:TRACe1:STORe[:DATA] Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.PN(Ch).TRACe(Tr).STORe.MEMory

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.PN(Ch).TRACe(Tr).STORe.MEMory = *Value*

Description

This command saves active trace memory, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Saves active trace memory
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 71, "Failed to write file" 76, "Save failed"

Examples

```
Dim Var as String
SCPI.MMEMory.PN1.TRACe1.STORe.MEMory = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:PN[1-1]:TRACe[1-1]:STORe:MEMory

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:PN1:TRACe1:STORe:MEMory Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.PN(Ch).TRACe(Tr).STORe.SPURious.OSSPur.FLISt

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.PN(Ch).TRACe(Tr).STORe.SPURious.OSSPur.FLISt = *Value*

Description

This command exports Spurious Table, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Exports Spurious Table
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 71, "Failed to write file"

Examples

```
Dim Var as String
SCPI.MMEMory.PN1.TRACe1.STORe.SPURious.OSSPur.FLISt = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:PN[1-1]:TRACe[1-1]:STORe:SPURious:OSSPur:FLISt

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:PN1:TRACe1:STORe:SPURious:OSSPur:FLISt Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.PS(Ch).TRACe(Tr).LOAD.DATA

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.PS(Ch).TRACe(Tr).LOAD.DATA = *Value*

Description

This command recalls data trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Recalls data trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-256, "File name not found" -257, "File name error" 70, "Failed to read file"

Examples

```
Dim Var as String
Var = "Sample_Text"
SCPI.MMEMory.PS1.TRACe1.LOAD.DATA = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:PS[1-1]:TRACe[1-1]:LOAD:DATA

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:PS1:TRACe1:LOAD:DATA Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.PS(Ch).TRACe(Tr).LOAD.LIMit.LOWer

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.PS(Ch).TRACe(Tr).LOAD.LIMit.LOWer = *Value*

Description

This command imports lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Lower limit line
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 70, "Failed to read file"

Examples

```
Dim Var as String
Var = "Sample_Text"
SCPI.MMEMory.PS1.TRACe1.LOAD.LIMit.LOWer = Var
```

Related Objects

SCPI.MMEMory.PS(Ch).TRACe(Tr).LOAD.LIMit.UPPer

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:PS[1-1]:TRACe[1-1]:LOAD:LIMit:LOWer

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:PS1:TRACe1:LOAD:LIMit:LOWer Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.PS(Ch).TRACe(Tr).LOAD.LIMit.UPPer

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.PS(Ch).TRACe(Tr).LOAD.LIMit.UPPer = *Value*

Description

This command imports upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Imports upper limit line
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 70, "Failed to read file"

Examples

```
Dim Var as String
Var = "Sample_Text"
SCPI.MMEMory.PS1.TRACe1.LOAD.LIMit.UPPer = Var
```

Related Objects

SCPI.MMEMory.PS(Ch).TRACe(Tr).LOAD.LIMit.LOWer

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:PS[1-1]:TRACe[1-1]:LOAD:LIMit:UPPer

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:PS1:TRACe1:LOAD:LIMit:UPPer Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.PS(Ch).TRACe(Tr).LOAD.MEMory

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.PS(Ch).TRACe(Tr).LOAD.MEMory = *Value*

Description

This command recalls memory trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Recalls memory trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-256, "File name not found" -257, "File name error" 70, "Failed to read file"

Examples

```
Dim Var as String
Var = "Sample_Text"
SCPI.MMEMory.PS1.TRACe1.LOAD.MEMory = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:PS[1-1]:TRACe[1-1]:LOAD:MEMory

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:PS1:TRACe1:LOAD:MEMory Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.PS(Ch).TRACe(Tr).LOAD.SPURious.OSSPur.FLISt

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.PS(Ch).TRACe(Tr).LOAD.SPURious.OSSPur.FLISt = *Value*

Description

This command imports Spurious Table, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Imports Spurious Table
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-256, "File name not found" -257, "File name error" 70,

	"Failed to read file"

Examples

```
Dim Var as String
Var = "Sample_Text"
SCPI.MMEMory.PS1.TRACe1.LOAD.SPURious.OSSPur.FLISt = Var
```

Related Objects

SCPI.MMEMory.PS(Ch).TRACe(Tr).LOAD.SPURious.THReshold

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:PS[1-1]:TRACe[1-1]:LOAD:SPURious:OSSPur:FLISt

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:PS1:TRACe1:LOAD:SPURious:OSSPur:FLISt Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.PS(Ch).TRACe(Tr).LOAD.SPURious.THReshold

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.PS(Ch).TRACe(Tr).LOAD.SPURious.THReshold = *Value*

Description

This command imports threshold table, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Imports threshold table
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 70,

	"Failed to read file"

Examples

```
Dim Var as String
Var = "Sample_Text"
SCPI.MMEMory.PS1.TRACe1.LOAD.SPURious.THReshold = Var
```

Related Objects

SCPI.MMEMory.PS(Ch).TRACe(Tr).LOAD.SPURious.OSSPur.FLISt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:PS[1-1]:TRACe[1-1]:LOAD:SPURious:THReshold

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:PS1:TRACe1:LOAD:SPURious:THReshold Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.PS(Ch).TRACe(Tr).STORe.DATA

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.PS(Ch).TRACe(Tr).STORe.DATA = *Value*

Description

This command saves active trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Saves active trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 71, "Failed to write file" 76, "Save failed"

Examples

```
Dim Var as String
Var = "Sample_Text"
SCPI.MMEMory.PS1.TRACe1.STORe.DATA = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:PS[1-1]:TRACe[1-1]:STORe[:DATA]

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:PS1:TRACe1:STORe[:DATA] Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.PS(Ch).TRACe(Tr).STORe.MEMory

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.PS(Ch).TRACe(Tr).STORe.MEMory = *Value*

Description

This command saves active trace memory, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Saves active trace memory
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 71, "Failed to write file" 76, "Save failed"

Examples

```
Dim Var as String
Var = "Sample_Text"
SCPI.MMEMory.PS1.TRACe1.STORe.MEMory = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:PS[1-1]:TRACe[1-1]:STORe:MEMory

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:PS1:TRACe1:STORe:MEMory Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.PS(Ch).TRACe(Tr).STORe.SPURious.OSSPur.FLISt

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.PS(Ch).TRACe(Tr).STORe.SPURious.OSSPur.FLISt = *Value*

Description

This command exports Spurious Table, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Exports Spurious Table
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 71, "Failed to write file"

Examples

```
Dim Var as String
Var = "Sample_Text"
SCPI.MMEMory.PS1.TRACe1.STORe.SPURious.OSSPur.FLISt = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:PS[1-1]:TRACe[1-1]:STORe:SPURious:OSSPur:FLISt

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:PS1:TRACe1:STORe:SPURious:OSSPur:FLISt Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.SP(Ch).TRACe(Tr).LOAD.DATA

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.SP(Ch).TRACe(Tr).LOAD.DATA = *Value*

Description

This command recalls data trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Recalls data trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-256, "File name not found" -257, "File name error" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.SP1.TRACe1.LOAD.DATA = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:SP[1-1]:TRACe[1-1]:LOAD:DATA

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:SP1:TRACe1:LOAD:DATA Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.SP(Ch).TRACe(Tr).LOAD.LIMit.LOWer

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.SP(Ch).TRACe(Tr).LOAD.LIMit.LOWer = *Value*

Description

This command imports lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Imports lower limit line
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.SP1.TRACe1.LOAD.LIMit.LOWer = Var
```

Related Objects

SCPI.MMEMory.SP(Ch).TRACe(Tr).LOAD.LIMit.UPPer

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:SP[1-1]:TRACe[1-1]:LOAD:LIMit:LOWer

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:SP1:TRACe1:LOAD:LIMit:LOWer Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.SP(Ch).TRACe(Tr).LOAD.LIMit.UPPer

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.SP(Ch).TRACe(Tr).LOAD.LIMit.UPPer = *Value*

Description

This command imports upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Imports upper limit line
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.SP1.TRACe1.LOAD.LIMit.UPPer = Var
```

Related Objects

SCPI.MMEMory.SP(Ch).TRACe(Tr).LOAD.LIMit.LOWer

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:SP[1-1]:TRACe[1-1]:LOAD:LIMit:UPPer

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:SP1:TRACe1:LOAD:LIMit:UPPer Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.SP(Ch).TRACe(Tr).LOAD.MEMory

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.SP(Ch).TRACe(Tr).LOAD.MEMory = *Value*

Description

This command recalls memory trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Recalls memory trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-256, "File name not found" -257, "File name error" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.SP1.TRACe1.LOAD.MEMory = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:SP[1-1]:TRACe[1-1]:LOAD:MEMory

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:SP1:TRACe1:LOAD:MEMory Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.SP(Ch).TRACe(Tr).STORe.DATA

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.SP(Ch).TRACe(Tr).STORe.DATA = *Value*

Description

This command saves active trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Saves active trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 71, "Failed to write file" 76, "Save failed"

Examples

```
Dim Var as String
SCPI.MMEMory.SP1.TRACe1.STORe.DATA = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:SP[1-1]:TRACe[1-1]:STORe[:DATA]

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:SP1:TRACe1:STORe[:DATA] Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.SP(Ch).TRACe(Tr).STORe.MEMory

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.SP(Ch).TRACe(Tr).STORe.MEMory = *Value*

Description

This command saves active trace memory, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Saves active trace memory
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 71, "Failed to write file" 76, "Save failed"

Examples

```
Dim Var as String
SCPI.MMEMory.SP1.TRACe1.STORe.MEMory = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:SP[1-1]:TRACe[1-1]:STORe:MEMory

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:SP1:TRACe1:STORe:MEMory Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.STORe.IMAGe

Object Type

Property (**Write Only**)

Syntax

SCPI.MMEMory.STORe.IMAGe = *File*

Description

This command saves the display image on the LCD display at the execution of the object into a file in the bitmap (extension ".bmp") or portable network graphics (extension ".png") format.

NOTE When saving the E5052B measurement screen, execute the VBA program with the Visual Basic editor closed. For more information, see Running_a_Program_from_the_E5052B_Measurement_Screen.

NOTE Specify the file name with the extension When you use directory names and file name, separate them with "\ " (back slash), or "/" (slash).

Variable

Parameter	*File*
Description	File name in which you want to save the display image on the LCD display (extension ".bmp" or ".png")
Data Type	Character string type (String)
Range	254 characters or less
Preset Value	-
Unit	-
Resolution	-

Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 71, "Failed to write file"

NOTE If a file with the same name as the specified file name exists, its contents are overwritten.

Examples

SCPI.MMEMory.STORe.IMAGe = "f:\image01.bmp"

SCPI.MMEMory.STORe.IMAGe = "test/image01.png"

Equivalent Key

System > **Dump Screen Image**

When performing the operation from the front panel, the image on the LCD display memorized in the volatile memory (clipboard) (the image on the LCD display when the **Capture** (**System**) key is pressed) is saved.

If no image is memorized in the clipboard, in the same way as the SCPI.MMEMory.STORe.IMAGe object, the image on the LCD display at the execution is memorized in the clipboard and then it is saved.

Equivalent SCPI Command

Syntax

:MMEMory:STORe:IMAGe <string>

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:STORe:IMAGe ""Result/Image01.bmp""""
20 OUTPUT 717;":MMEMory:STORe:IMAGe ""D:Image01.png""""
```

SCPI.MMEMory.STORe.PROGram

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.STORe.PROGram = *Value*

Description

This command saves VBA project.

Variable

Parameter	*Value*
Description	Saves VBA project
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 71, "Failed to write file" 78, "Save VBA program failed"

Examples

```
Dim Var as String
SCPI.MMEMory.STORe.PROGram = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:STORe:PROGram

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:STORe:PROGram Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.STORe.STATe

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.STORe.STATe = *Value*

Description

This command saves settings.

Variable

Parameter	*Value*
Description	Saves settings
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 71, "Failed to write file" 76, "Save failed"

Examples

```
Dim Var as String
SCPI.MMEMory.STORe.STATe = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:STORe:STATe

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:STORe:STATe Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.STORe.STYPe

Object Type

Property (**Read-Write**)

Syntax

SCPI.MMEMory.STORe.STYPe = *Value*

Value = SCPI.MMEMory.STORe.STYPe

Description

This command sets/gets E5052B state type.

Variable

Parameter	*Value*
Description	State type
Data Type	Character string type (String)
Range	STATe: Instrument States Only DSTate:Instrument States and Data
Preset Value	STATe
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as String
Var= "STATe"
SCPI.MMEMory.STORe.STYPe = Var
Var = SCPI.MMEMory.STORe.STYPe
```

Equivalent Key

PN Menu: **Save/Recall** > **Save State** > **Save Type**

SP Menu: **Save/Recall** > **Save State** > **Save Type**

FP Menu: **Save/Recall** > **Save State** > **Save Type**

TR Menu: **Save/Recall** > **Save State** > **Save Type**

AM Menu: **Save/Recall** > **Save State** > **Save Type**

BB Menu: **Save/Recall** > **Save State** > **Save Type**

PS Menu: **Save/Recall** > **Save State** > **Save Type**

USER Menu: **Save/Recall** > **Save State** > **Save Type**

Equivalent SCPI Command

Syntax

:MMEMory:STORe:STYPe {STATe|DSTate}

:MMEMory:STORe:STYPe?

Query Response

{STATe|DSTate} <newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:STORe:STYPe STATe"
20 OUTPUT 717;":MMEMory:STORe:STYPe ?"
30 ENTER 717;A$
```

SCPI.MMEMory.STORe.TRACe.AINFormation

Object Type

Property (**Read-Write**)

Syntax

SCPI.MMEMory.STORe.TRACe.AINFormation = *Value*

Value = SCPI.MMEMory.STORe.TRACe.AINFormation

Description

This command sets/gets trace/memory header.

NOTE This function is available when measurements mode are AM, PN and PS.

Variable

Parameter	*Value*
Description	trace/memory header
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.MMEMory.STORe.TRACe.AINFormation = Var
Var = SCPI.MMEMory.STORe.TRACe.AINFormation
```

Equivalent Key

AM Menu: **Save/Recall** > **Additional Info**

PN Menu: **Save/Recall** > **Additional Info**

PS Menu: **Save/Recall** > **Additional Info**

Equivalent SCPI Command

Syntax

:MMEMory:STORe:TRACe:AINFormation {ON|OFF|1|0}

:MMEMory:STORe:TRACe:AINFormation?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:STORe:TRACe:AINFormation 1"
20 OUTPUT 717;":MMEMory:STORe:TRACe:AINFormation?"
30 ENTER 717;A
```

SCPI.MMEMory.TR(Ch).TRACe(Tr).LOAD.DATA

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.TR(Ch).TRACe(Tr).LOAD.DATA = *Value*

Description

This command recalls data trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Recalls data trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-256, "File name not found" -257, "File name error" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.TR1.TRACe4.LOAD.DATA = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:TR[1-1]:TRACe[1-8]:LOAD:DATA

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:TR1:TRACe4:LOAD:DATA Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.TR(Ch).TRACe(Tr).LOAD.LIMit.LOWer

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.TR(Ch).TRACe(Tr).LOAD.LIMit.LOWer = *Value*

Description

This command import lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Import lower limit line
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.TR1.TRACe4.LOAD.LIMit.LOWer = Var
```

Related Objects

SCPI.MMEMory.TR(Ch).TRACe(Tr).LOAD.LIMit.UPPer

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:TR[1-1]:TRACe[1-8]:LOAD:LIMit:LOWer

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:TR1:TRACe4:LOAD:LIMit:LOWer Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.TR(Ch).TRACe(Tr).LOAD.LIMit.UPPer

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.TR(Ch).TRACe(Tr).LOAD.LIMit.UPPer = *Value*

Description

This command imports upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Imports upper limit line
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.TR1.TRACe4.LOAD.LIMit.UPPer = Var
```

Related Objects

SCPI.MMEMory.TR(Ch).TRACe(Tr).LOAD.LIMit.LOWer

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:TR[1-1]:TRACe[1-8]:LOAD:LIMit:UPPer

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:TR1:TRACe4:LOAD:LIMit:UPPer Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.TR(Ch).TRACe(Tr).LOAD.MEMory

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.TR(Ch).TRACe(Tr).LOAD.MEMory = *Value*

Description

This command recalls memory trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Recalls memory trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-256, "File name not found" -257, "File name error" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.TR1.TRACe4.LOAD.MEMory = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:TR[1-1]:TRACe[1-8]:LOAD:MEMory

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:TR1:TRACe4:LOAD:MEMory Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.TR(Ch).TRACe(Tr).STORe.DATA

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.TR(Ch).TRACe(Tr).STORe.DATA = *Value*

Description

This command saves active trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Saves active trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 71, "Failed to write file" 76, "Save failed"

Examples

```
Dim Var as String
SCPI.MMEMory.TR1.TRACe4.STORe.DATA = Var
```

Related Objects

SCPI.MMEMory.TR(Ch).TRACe(Tr).STORe.MEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:TR[1-1]:TRACe[1-8]:STORe[:DATA]

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:TR1:TRACe4:STORe[:DATA] Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.TR(Ch).TRACe(Tr).STORe.MEMory

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.TR(Ch).TRACe(Tr).STORe.MEMory = *Value*

Description

This command saves active trace memory, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Saves active trace memory
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 71, "Failed to write file" 76, "Save failed"

Examples

```
Dim Var as String
SCPI.MMEMory.TR1.TRACe4.STORe.MEMory = Var
```

Related Objects

SCPI.MMEMory.TR(Ch).TRACe(Tr).STORe.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:TR[1-1]:TRACe[1-8]:STORe:MEMory

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:TR1:TRACe4:STORe:MEMory Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.USER(Ch).TRACe(Tr).LOAD.DATA

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.USER(Ch).TRACe(Tr).LOAD.DATA = *Value*

Description

This command recalls data trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Recalls data trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-256, "File name not found" -257, "File name error" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.USER1.TRACe8.LOAD.DATA = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:USER[1-1]:TRACe[1-8]:LOAD:DATA

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:USER1:TRACe8:LOAD:DATA Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.USER(Ch).TRACe(Tr).LOAD.LIMit.LOWer

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.USER(Ch).TRACe(Tr).LOAD.LIMit.LOWer = *Value*

Description

This command imports lower limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Imports lower limit line
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.USER1.TRACe8.LOAD.LIMit.LOWer = Var
```

Related Objects

SCPI.MMEMory.USER(Ch).TRACe(Tr).LOAD.LIMit.UPPer

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:USER[1-1]:TRACe[1-8]:LOAD:LIMit:LOWer

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:USER1:TRACe8:LOAD:LIMit:LOWer Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.USER(Ch).TRACe(Tr).LOAD.LIMit.UPPer

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.USER(Ch).TRACe(Tr).LOAD.LIMit.UPPer = *Value*

Description

This command imports upper limit line, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Imports upper limit line
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" -256, "File name not found" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.USER1.TRACe8.LOAD.LIMit.UPPer = Var
```

Related Objects

SCPI.MMEMory.USER(Ch).TRACe(Tr).LOAD.LIMit.LOWer

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:USER[1-1]:TRACe[1-8]:LOAD:LIMit:UPPer

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:USER1:TRACe8:LOAD:LIMit:UPPer Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.USER(Ch).TRACe(Tr).LOAD.MEMory

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.USER(Ch).TRACe(Tr).LOAD.MEMory = *Value*

Description

This command recalls memory trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Recalls memory trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-256, "File name not found" -257, "File name error" 70, "Failed to read file"

Examples

```
Dim Var as String
SCPI.MMEMory.USER1.TRACe8.LOAD.MEMory = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:USER[1-1]:TRACe[1-8]:LOAD:MEMory

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:USER1:TRACe8:LOAD:MEMory Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.USER(Ch).TRACe(Tr).STORe.DATA

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.USER(Ch).TRACe(Tr).STORe.DATA = *Value*

Description

This command saves active trace data, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Saves active trace data
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 71, "Failed to write file" 76, "Save failed"

Examples

```
Dim Var as String
SCPI.MMEMory.USER1.TRACe8.STORe.DATA = Var
```

Related Objects

SCPI.MMEMory.USER(Ch).TRACe(Tr).STORe.MEMory

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:USER[1-1]:TRACe[1-8]:STORe[:DATA]

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:USER1:TRACe8:STORe[:DATA] Sample_Text"
20 ENTER 717;A$
```

SCPI.MMEMory.USER(Ch).TRACe(Tr).STORe.MEMory

Object Type

Method (**Write Only**)

Syntax

SCPI.MMEMory.USER(Ch).TRACe(Tr).STORe.MEMory = *Value*

Description

This command saves active trace memory, for the selected trace *Tr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Saves active trace memory
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-257, "File name error" 71, "Failed to write file" 76, "Save failed"

Examples

```
Dim Var as String
SCPI.MMEMory.USER1.TRACe8.STORe.MEMory = Var
```

Related Objects

SCPI.MMEMory.USER(Ch).TRACe(Tr).STORe.DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:MMEMory:USER[1-1]:TRACe[1-8]:STORe:MEMory

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":MMEMory:USER1:TRACe8:STORe:MEMory Sample_Text"
20 ENTER 717;A$
```

Program

SCPI.PROGram.CATalog

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.PROGram.CATalog

Description

This command gets the list of all defined programs.

Variable

Parameter	*Value*
Description	Gets the list of all defined programs
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as String
Var = SCPI.PROGram.CATalog
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:PROGram:CATalog?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":PROGram:CATalog ?"
20 ENTER 717;A$
```

SCPI.PROGram.COM.EVENt

Object Type

Property (**Read-Write**)

Syntax

SCPI.PROGram.COM.EVENt = *Value*

Value = SCPI.PROGram.COM.EVENt

Description

This command sets/gets VBA callback.

Variable

Parameter	*Value*
Description	VBA callback
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.PROGram.COM.EVENt = Var
Var = SCPI.PROGram.COM.EVENt
```

Equivalent Key

PN Menu: **Macro Setup** > **E5052 Event**

SP Menu: **Macro Setup** > **E5052 Event**

FP Menu: **Macro Setup** > **E5052 Event**

TR Menu: **Macro Setup** > **E5052 Event**

AM Menu: **Macro Setup** > **E5052 Event**

BB Menu: **Macro Setup** > **E5052 Event**

PS Menu: **Macro Setup** > **E5052 Event**

USER Menu: **Macro Setup** > **E5052 Event**

Equivalent SCPI Command

Syntax

:PROGram:COM:EVENt {ON|OFF|1|0}

:PROGram:COM:EVENt?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":PROGram:COM:EVENt 1"
20 OUTPUT 717;":PROGram:COM:EVENt ?"
30 ENTER 717;A
```

SCPI.PROGram.COMMand

Object Type

Method (**Write Only**)

Syntax

SCPI.PROGram.COMMand = *Value*

Description

This command sends command to VBA.

Variable

Parameter	*Value*
Description	Sends command to VBA
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	92, "Program not running" 93, "Program command error"

Examples

```
Dim Var as String
SCPI.PROGram.COMMand = Var
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:PROGram:COMMand

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":PROGram:COMMand Sample_Text"
20 ENTER 717;A$
```

SCPI.PROGram.QUERy_Q cmd, ret

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.PROGram.QUERy_Q cmd, ret

Description

This command gets query to VBA.

Variable

Parameter	*cmd*
Description	Query to VBA
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	92, "Program not running"

Parameter	*ret*
Description	Query to VBA
Data Type	Character string type (String)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	93, "Program command error"

Examples

```
Dim Var as Variant
Var = SCPI.PROGram.QUERy_Q cmd, ret
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:PROGram:QUERy?

Query Response

{254 chars -} <newline><^END>

Example of use

```
10 OUTPUT 717;":PROGram:QUERy ?"
20 ENTER 717;A$
```

SCPI.PROGram.SELected.NAME

Object Type

Property (**Read-Write**)

Syntax

SCPI.PROGram.SELected.NAME = *Value*

Value = SCPI.PROGram.SELected.NAME

Description

This command sets/gets the name of the program to be selected.

Variable

Parameter	*Value*
Description	Name of the program to be selected
Data Type	Character string type (String)
Range	254 chars
Preset Value	"E5052.Module1.Main"
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as String
SCPI.PROGram.SELected.NAME = Var
Var = SCPI.PROGram.SELected.NAME
```

Related Objects

SCPI.PROGram.SELected.STATe

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:PROGram:SELected:NAME

:PROGram:SELected:NAME?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":PROGram:SELected:NAME Sample_Text"
20 OUTPUT 717;":PROGram:SELected:NAME ?"
30 ENTER 717;A$
```

SCPI.PROGram.SELected.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.PROGram.SELected.STATe = *Value*

Value = SCPI.PROGram.SELected.STATe

Description

This command sets/gets the state of a selected program.

Variable

Parameter	*Value*
Description	State of a selected program
Data Type	Character string type (String)
Range	STOP|RUN
Preset Value	STOP
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-286, "Program runtime error" -284, "Program currently running" -282, "Illegal program name" 90, "Failed to stop

	program"

Examples

```
Dim Var as String
Var= "STOP"
SCPI.PROGram.SELected.STATe = Var
Var = SCPI.PROGram.SELected.STATe
```

Related Objects

SCPI.PROGram.SELected.NAME

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:PROGram:SELected:STATe {STOP|RUN}

:PROGram:SELected:STATe?

Query Response

{STOP|RUN} <newline><^END>

Example of use

```
10 OUTPUT 717;":PROGram:SELected:STATe STOP"
20 OUTPUT 717;":PROGram:SELected:STATe ?"
30 ENTER 717;A$
```

SCPI.PROGram.SKEY.ITEM(1-8).ENABle

Object Type

Property (**Read-Write**)

Syntax

SCPI.PROGram.SKEY.ITEM(1-8).ENABle = *Value*

Value = SCPI.PROGram.SKEY.ITEM(1-8).ENABle

Description

This command enable/disable user defined key label.

Variable

Parameter	*Value*
Description	User defined key label
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.PROGram.SKEY.ITEM8.ENABle = Var
Var = SCPI.PROGram.SKEY.ITEM8.ENABle
```

Related Objects

SCPI.PROGram.SKEY.ITEM(1-8).ENABle

SCPI.PROGram.SKEY.ITEM(1-8).IMMediate

SCPI.PROGram.SKEY.ITEM(1-8).LABel

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:PROGram:SKEY:ITEM[1-8]:ENABle {ON|OFF|1|0}

:PROGram:SKEY:ITEM[1-8]:ENABle?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":PROGram:SKEY:ITEM8:ENABle 1"
20 OUTPUT 717;":PROGram:SKEY:ITEM8:ENABle ?"
30 ENTER 717;A
```

SCPI.PROGram.SKEY.ITEM(1-8).IMMediate

Object Type

Method (**Write Only**)

Syntax

SCPI.PROGram.SKEY.ITEM(1-8).IMMediate

Description

This command executes user defined key. This command generates an error if either the program is not running, event is not enabled or if the item is not enabled.

Examples

SCPI.PROGram.SKEY.ITEM8.IMMediate

Related Objects

SCPI.PROGram.SKEY.ITEM(1-8).ENABle

SCPI.PROGram.SKEY.ITEM(1-8).LABel

Equivalent Key

PN Menu: **Macro Setup** > **User Menu** > **User Label 1**

SP Menu: **Macro Setup** > **User Menu** > **User Label 1**

FP Menu: **Macro Setup** > **User Menu** > **User Label 1**

TR Menu: **Macro Setup** > **User Menu** > **User Label 1**

AM Menu: **Macro Setup** > **User Menu** > **User Label 1**

BB Menu: **Macro Setup** > **User Menu** > **User Label 1**

USER Menu: **Macro Setup** > **User Menu** > **User Label 1**

Equivalent SCPI Command

Syntax

:PROGram:SKEY:ITEM[1-8]:IMMediate

Example of use

```
10 OUTPUT 717;":PROGram:SKEY:ITEM8:IMMediate"
```

SCPI.PROGram.SKEY.ITEM(1-8).LABel

Object Type

Property (**Read-Write**)

Syntax

SCPI.PROGram.SKEY.ITEM(1-8).LABel = *Value*

Value = SCPI.PROGram.SKEY.ITEM(1-8).LABel

Description

This command sets/gets user defined key label.

Variable

Parameter	*Value*
Description	User defined key label
Data Type	Character string type (String)
Range	30 chars
Preset Value	"User Label 1"
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as String
Var= "STOP"
SCPI.PROGram.SKEY.ITEM8.LABel = Var
Var = SCPI.PROGram.SKEY.ITEM8.LABel
```

Related Objects

SCPI.PROGram.SKEY.ITEM(1-8).ENABle

SCPI.PROGram.SKEY.ITEM(1-8).IMMediate

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:PROGram:SKEY:ITEM[1-8]:LABel {30 chars}

:PROGram:SKEY:ITEM[1-8]:LABel?

Query Response

{30 chars} <newline><^END>

Example of use

```
10 OUTPUT 717;":PROGram:SKEY:ITEM8:LABel Sample_Text"
20 OUTPUT 717;":PROGram:SKEY:ITEM8:LABel ?"
30 ENTER 717;A$
```

SCPI.PROGram.VARiable.ARRay(1-10).DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.PROGram.VARiable.ARRay(1-10).DATA = *Value*

Value = SCPI.PROGram.VARiable.ARRay(1-10).DATA

Description

This command sets/gets user defined array data.

Variable

Parameter	*Value*
Description	User defined array data
Data Type	Variant type Array (Range)
Range	1...1601
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var(1 to 1601) as Variant
SCPI.PROGram.VARiable.ARRay10.DATA = Var
Var = SCPI.PROGram.VARiable.ARRay10.DATA
```

Related Objects

SCPI.PROGram.VARiable.ARRay(1-10).POINts

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:PROGram:VARiable:ARRay[1-10]:DATA {1...1601}

:PROGram:VARiable:ARRay[1-10]:DATA?

Query Response

{1...1601} <newline><^END>

Example of use

```
10 Dim A(1:1601)
20 OUTPUT 717;":PROGram:VARiable:ARRay10:DATA 1,1601"
30 OUTPUT 717;":PROGram:VARiable:ARRay10:DATA ?"
40 ENTER 717;A(*)
```

SCPI.PROGram.VARiable.ARRay(1-10).POINts

Object Type

Property (**Read-Write**)

Syntax

SCPI.PROGram.VARiable.ARRay(1-10).POINts = *Value*

Value = SCPI.PROGram.VARiable.ARRay(1-10).POINts

Description

This command sets/gets number of points in a user defined array.

Variable

Parameter	*Value*
Description	Number of points in a user defined array
Data Type	Long integer type (Long)
Range	2 ~ 1601
Preset Value	1601
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 2
SCPI.PROGram.VARiable.ARRay10.POINts = Var
Var = SCPI.PROGram.VARiable.ARRay10.POINts
```

Related Objects

SCPI.PROGram.VARiable.ARRay(1-10).DATA

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:PROGram:VARiable:ARRay[1-10]:POINts {2 ~ 1601}

:PROGram:VARiable:ARRay[1-10]:POINts?

Query Response

{2 ~ 1601} <newline><^END>

Example of use

```
10 OUTPUT 717;":PROGram:VARiable:ARRay10:POINts 2 "
20 OUTPUT 717;":PROGram:VARiable:ARRay10:POINts ?"
30 ENTER 717;A
```

SCPI.PROGram.VARiable.DOUBle(1-10)

Object Type

Property (**Read-Write**)

Syntax

SCPI.PROGram.VARiable.DOUBle(1-10) = *Value*

Value = SCPI.PROGram.VARiable.DOUBle(1-10)

Description

This command sets/gets user defined 64bit floating variable.

Variable

Parameter	*Value*
Description	User defined 64bit floating variable
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
SCPI.PROGram.VARiable.DOUBle10 = Var
Var = SCPI.PROGram.VARiable.DOUBle10
```

Related Objects

SCPI.PROGram.VARiable.INTeger(1-10)

SCPI.PROGram.VARiable.STRing(1-10)

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:PROGram:VARiable:DOUBle[1-10]

:PROGram:VARiable:DOUBle[1-10]?

Query Response

{Double} <newline><^END>

Example of use

```
10 OUTPUT 717;":PROGram:VARiable:DOUBle10 0"
20 OUTPUT 717;":PROGram:VARiable:DOUBle10 ?"
30 ENTER 717;A
```

SCPI.PROGram.VARiable.INTeger(1-10)

Object Type

Property (**Read-Write**)

Syntax

SCPI.PROGram.VARiable.INTeger(1-10) = *Value*

Value = SCPI.PROGram.VARiable.INTeger(1-10)

Description

This command sets/gets user defined integer variable.

Variable

Parameter	*Value*
Description	User defined integer variable
Data Type	Long integer type (Long)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
SCPI.PROGram.VARiable.INTeger10 = Var
Var = SCPI.PROGram.VARiable.INTeger10
```

Related Objects

SCPI.PROGram.VARiable.DOUBle(1-10)

SCPI.PROGram.VARiable.STRing(1-10)

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:PROGram:VARiable:INTeger[1-10]

:PROGram:VARiable:INTeger[1-10]?

Query Response

{Long} <newline><^END>

Example of use

```
10 OUTPUT 717;":PROGram:VARiable:INTeger10 -"
20 OUTPUT 717;":PROGram:VARiable:INTeger10 ?"
30 ENTER 717;A
```

SCPI.PROGram.VARiable.STRing(1-10)

Object Type

Property (**Read-Write**)

Syntax

SCPI.PROGram.VARiable.STRing(1-10) = *Value*

Value = SCPI.PROGram.VARiable.STRing(1-10)

Description

This command sets/gets user defined string.

Variable

Parameter	*Value*
Description	User defined string
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as String
SCPI.PROGram.VARiable.STRing10 = Var
Var = SCPI.PROGram.VARiable.STRing10
```

Related Objects

SCPI.PROGram.VARiable.DOUBle(1-10)

SCPI.PROGram.VARiable.INTeger(1-10)

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:PROGram:VARiable:STRing[1-10]

:PROGram:VARiable:STRing[1-10]?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":PROGram:VARiable:STRing10 Sample_Text"
20 OUTPUT 717;":PROGram:VARiable:STRing10 ?"
30 ENTER 717;A$
```

Sense

SCPI.SENSe.AM(Ch).ASET

Object Type

Method (**Write Only**)

Syntax

SCPI.SENSe.AM(Ch).ASET

Description

This command sets Attn, Freq Band and IF Gain Auto Set, for the selected channel *Ch*.

Examples

SCPI.SENSe.AM1.ASET

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SENSe:AM[1-1]:ASET

Example of use

10 OUTPUT 717;":SENSe:AM1:ASET"

SCPI.SENSe.AM(Ch).AVERage.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.SENSe.AM(Ch).AVERage.CLEar

Description

This command sets average clear, for the selected channel *Ch*.

Examples

SCPI.SENSe.AM1.AVERage.CLEar

Related Objects

SCPI.SENSe.AM(Ch).AVERage.COUNt

SCPI.SENSe.AM(Ch).AVERage.STATe

Equivalent Key

AM Menu: **Average** > **Averaging Restart**

Equivalent SCPI Command

Syntax

:SENSe:AM[1-1]:AVERage:CLEar

Example of use

```
10 OUTPUT 717;":SENSe:AM1:AVERage:CLEar"
```

SCPI.SENSe.AM(Ch).AVERage.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.AM(Ch).AVERage.COUNt = *Value*

Value = SCPI.SENSe.AM(Ch).AVERage.COUNt

Description

This command sets/gets average count, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Average count
Data Type	Long integer type (Long)
Range	1 ~ 999
Preset Value	16
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.SENSe.AM1.AVERage.COUNt = Var
Var = SCPI.SENSe.AM1.AVERage.COUNt
```

Related Objects

SCPI.SENSe.AM(Ch).AVERage.CLEar

SCPI.SENSe.AM(Ch).AVERage.STATe

Equivalent Key

AM Menu: **Average** > **Avg Factor**

Equivalent SCPI Command

Syntax

:SENSe:AM[1-1]:AVERage:COUNt {1 ~ 999}

:SENSe:AM[1-1]:AVERage:COUNt?

Query Response

{1 ~ 999} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:AM1:AVERage:COUNt 1 "
20 OUTPUT 717;":SENSe:AM1:AVERage:COUNt ?"
30 ENTER 717;A
```

SCPI.SENSe.AM(Ch).AVERage.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.AM(Ch).AVERage.STATe = *Value*

Value = SCPI.SENSe.AM(Ch).AVERage.STATe

Description

This command sets/gets average ON/OFF, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Average (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.SENSe.AM1.AVERage.STATe = Var
Var = SCPI.SENSe.AM1.AVERage.STATe
```

Related Objects

SCPI.SENSe.AM(Ch).AVERage.CLEar

SCPI.SENSe.AM(Ch).AVERage.COUNt

Equivalent Key

AM Menu: **Average** > **Averaging**

Equivalent SCPI Command

Syntax

:SENSe:AM[1-1]:AVERage:STATe {ON|OFF|1|0}

:SENSe:AM[1-1]:AVERage:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:AM1:AVERage:STATe 1"
20 OUTPUT 717;":SENSe:AM1:AVERage:STATe ?"
30 ENTER 717;A
```

SCPI.SENSe.AM(Ch).CORRelation.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.AM(Ch).CORRelation.COUNt = *Value*

Value = SCPI.SENSe.AM(Ch).CORRelation.COUNt

Description

This command sets/gets correlation count, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Correlation count
Data Type	Long integer type (Long)
Range	1 ~ 10000
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.SENSe.AM1.CORRelation.COUNt = Var
Var = SCPI.SENSe.AM1.CORRelation.COUNt
```

Equivalent Key

AM Menu: **Average** > **Correlation**

Equivalent SCPI Command

Syntax

:SENSe:AM[1-1]:CORRelation:COUNt {1 ~ 10000}

:SENSe:AM[1-1]:CORRelation:COUNt?

Query Response

{1 ~ 10000} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:AM1:CORRelation:COUNt 1 "
20 OUTPUT 717;":SENSe:AM1:CORRelation:COUNt ?"
30 ENTER 717;A
```

SCPI.SENSe.AM(Ch).DCONverter.FREQuency

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.AM(Ch).DCONverter.FREQuency = *Value*

Value = SCPI.SENSe.AM(Ch).DCONverter.FREQuency

Description

This command sets/gets downconverter input frequency (carrier nominal frequency), for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Downconverter input frequency (carrier nominal frequency)
Data Type	Double precision floating point type (Double)
Range	3G ~ 26.5G
Preset Value	3G
Unit	Hz
Resolution	100m
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
Var= 3000000000
SCPI.SENSe.AM1.DCONverter.FREQuency = Var
Var = SCPI.SENSe.AM1.DCONverter.FREQuency
```

Related Objects

SCPI.SENSe.AM(Ch).DCONverter.SSEarch.EXECute

Equivalent Key

AM Menu: **Setup** > **Nominal Frequency**

Equivalent SCPI Command

Syntax

:SENSe:AM[1-1]:DCONverter:FREQuency {3G ~ 26.5G}

:SENSe:AM[1-1]:DCONverter:FREQuency?

Query Response

{3G ~ 26.5G} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:AM1:DCONverter:FREQuency 3000000000"
20 OUTPUT 717;":SENSe:AM1:DCONverter:FREQuency ?"
30 ENTER 717;A
```

SCPI.SENSe.AM(Ch).DCONverter.SSEarch.EXECute

Object Type

Method (**Write Only**)

Syntax

SCPI.SENSe.AM(Ch).DCONverter.SSEarch.EXECute

Description

This command searches carrier signal, for the selected channel *Ch*.

Examples

SCPI.SENSe.AM1.DCONverter.SSEarch.EXECute

Related Objects

SCPI.SENSe.AM(Ch).DCONverter.FREQuency

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SENSe:AM[1-1]:DCONverter:SSEarch:EXECute

Example of use

```
10 OUTPUT 717;":SENSe:AM1:DCONverter:SSEarch:EXECute"
```

SCPI.SENSe.AM(Ch).FAST

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.AM(Ch).FAST = *Value*

Value = SCPI.SENSe.AM(Ch).FAST

Description

This command sets/gets fast measurement mode, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Fast measurement mode
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.SENSe.AM1.FAST = Var
Var = SCPI.SENSe.AM1.FAST
```

Equivalent Key

AM Menu: **Setup** > **Fast Measurement**

Equivalent SCPI Command

Syntax

:SENSe:AM[1-1]:FAST {ON|OFF|1|0}

:SENSe:AM[1-1]:FAST?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:AM1:FAST 1"
20 OUTPUT 717;":SENSe:AM1:FAST ?"
30 ENTER 717;A
```

SCPI.SENSe.AM(Ch).FBANd

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.AM(Ch).FBANd = *Value*

Value = SCPI.SENSe.AM(Ch).FBANd

Description

This command sets/gets frequency band, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Frequency band
Data Type	Character string type (String)
Range	BAND2|BAND3|BAND4|BAND5|BAND6
Preset Value	BAND4
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "BAND2"
SCPI.SENSe.AM1.FBANd = Var
Var = SCPI.SENSe.AM1.FBANd
```

Equivalent Key

AM Menu: **Setup** > **Frequency Band**

Equivalent SCPI Command

Syntax

:SENSe:AM[1-1]:FBANd {BAND2|BAND3|BAND4|BAND5|BAND6}

:SENSe:AM[1-1]:FBANd?

Query Response

{BAND2|BAND3|BAND4|BAND5|BAND6} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:AM1:FBANd BAND2"
20 OUTPUT 717;":SENSe:AM1:FBANd ?"
30 ENTER 717;A$
```

SCPI.SENSe.AM(Ch).FREQuency.STARt

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.AM(Ch).FREQuency.STARt = *Value*

Value = SCPI.SENSe.AM(Ch).FREQuency.STARt

Description

This command sets/gets start frequency, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Start frequency
Data Type	Double precision floating point type (Double)
Range	1|10|100|1k
Preset Value	1k
Unit	Hz
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SENSe.AM1.FREQuency.STARt = Var
Var = SCPI.SENSe.AM1.FREQuency.STARt
```

Related Objects

SCPI.SENSe.AM(Ch).FREQuency.STOP

Equivalent Key

AM Menu: **Start** > **1Hz**

AM Menu: **Start** > **10Hz**

AM Menu: **Start** > **100Hz**

AM Menu: **Start** > **1kHz**

AM Menu: **Marker To** > **Marker** > **Start**

Equivalent SCPI Command

Syntax

:SENSe:AM[1-1]:FREQuency:STARt {1|10|100|1k}

:SENSe:AM[1-1]:FREQuency:STARt?

Query Response

{1|10|100|1k} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:AM1:FREQuency:STARt 0"
20 OUTPUT 717;":SENSe:AM1:FREQuency:STARt ?"
30 ENTER 717;A
```

SCPI.SENSe.AM(Ch).FREQuency.STOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.AM(Ch).FREQuency.STOP = *Value*

Value = SCPI.SENSe.AM(Ch).FREQuency.STOP

Description

This command sets/gets stop frequency, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Stop frequency
Data Type	Double precision floating point type (Double)
Range	100k|1M|5M|10M|20M|40M
Preset Value	10M
Unit	Hz
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 100000
SCPI.SENSe.AM1.FREQuency.STOP = Var
Var = SCPI.SENSe.AM1.FREQuency.STOP
```

Related Objects

SCPI.SENSe.AM(Ch).FREQuency.STARt

Equivalent Key

AM Menu: **Stop** > **100kHz**

AM Menu: **Stop** > **1MHz**

AM Menu: **Stop** > **5MHz**

AM Menu: **Stop** > **10MHz**

AM Menu: **Stop** > **20MHz**

AM Menu: **Stop** > **40MHz**

AM Menu: **Marker To** > **Marker** > **Stop**

Equivalent SCPI Command

Syntax

:SENSe:AM[1-1]:FREQuency:STOP {100k|1M|5M|10M|20M|40M}

:SENSe:AM[1-1]:FREQuency:STOP?

Query Response

{100k|1M|5M|10M|20M|40M} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:AM1:FREQuency:STOP 100000"
20 OUTPUT 717;":SENSe:AM1:FREQuency:STOP ?"
30 ENTER 717;A
```

SCPI.SENSe.AM(Ch).SEGTable.MEASurement.QUALity

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.AM(Ch).SEGTable.MEASurement.QUALity = *Value*

Value = SCPI.SENSe.AM(Ch).SEGTable.MEASurement.QUALity

Description

This command specifies the segment table to be used, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Specifies the segment table to be used
Data Type	Character string type (String)
Range	NORMal|FAST
Preset Value	NORMal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NORMal"
SCPI.SENSe.AM1.SEGTable.MEASurement.QUALity = Var
Var = SCPI.SENSe.AM1.SEGTable.MEASurement.QUALity
```

Equivalent Key

AM Menu: **Setup** > **Measurement Quality**

Equivalent SCPI Command

Syntax

:SENSe:AM[1-1]:SEGTable[:MEASurement][:QUALity] {NORMal|FAST}

:SENSe:AM[1-1]:SEGTable[:MEASurement][:QUALity]?

Query Response

{NORMal|FAST} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:AM1:SEGTable[:MEASurement][:QUALity] NORMal"
20 OUTPUT 717;":SENSe:AM1:SEGTable[:MEASurement][:QUALity] ?"
30 ENTER 717;A$
```

SCPI.SENSe.AM(Ch).SWEep.POINts

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.SENSe.AM(Ch).SWEep.POINts

Description

This command gets number of points, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Number of points
Data Type	Long integer type (Long)
Range	2 ~ 1601
Preset Value	517
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 2
Var = SCPI.SENSe.AM1.SWEep.POINts
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SENSe:AM[1-1]:SWEep:POINts?

Query Response

{2 ~ 1601} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:AM1:SWEep:POINts ?"
20 ENTER 717;A
```

SCPI.SENSe.ATTenuation.LEVel

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.ATTenuation.LEVel = *Value*

Value = SCPI.SENSe.ATTenuation.LEVel

Description

This command sets/gets Input Attenuator level on 5dB Step.

Variable

Parameter	*Value*
Description	Input Attenuator level on 5dB Step
Data Type	Double precision floating point type (Double)
Range	0 ~ 35
Preset Value	5
Unit	dB
Resolution	5
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SENSe.ATTenuation.LEVel = Var
Var = SCPI.SENSe.ATTenuation.LEVel
```

Equivalent Key

PN Menu: **Attenuator** > **Input Attenuator**

SP Menu: **Attenuator** > **Input Attenuator**

FP Menu: **Attenuator** > **Input Attenuator**

TR Menu: **Attenuator** > **Input Attenuator**

AM Menu: **Attenuator** > **Input Attenuator**

BB Menu: **Attenuator** > **Input Attenuator**

USER Menu: **Attenuator** > **Input Attenuator**

Equivalent SCPI Command

Syntax

:SENSe:ATTenuation:LEVel {0 ~ 35}

:SENSe:ATTenuation:LEVel?

Query Response

{0 ~ 35} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:ATTenuation:LEVel 0"
20 OUTPUT 717;":SENSe:ATTenuation:LEVel ?"
30 ENTER 717;A
```

SCPI.SENSe.BB(Ch).AVERage.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.SENSe.BB(Ch).AVERage.CLEar

Description

This command clears average, for the selected channel *Ch*.

Examples

SCPI.SENSe.BB1.AVERage.CLEar

Related Objects

SCPI.SENSe.BB(Ch).AVERage.COUNt

SCPI.SENSe.BB(Ch).AVERage.STATe

Equivalent Key

BB Menu: **Average** > **Averaging Restart**

Equivalent SCPI Command

Syntax

:SENSe:BB[1-1]:AVERage:CLEar

Example of use

```
10 OUTPUT 717;":SENSe:BB1:AVERage:CLEar"
```

SCPI.SENSe.BB(Ch).AVERage.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.BB(Ch).AVERage.COUNt = *Value*

Value = SCPI.SENSe.BB(Ch).AVERage.COUNt

Description

This command sets/gets average count, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Average count
Data Type	Long integer type (Long)
Range	1 ~ 999
Preset Value	16
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.SENSe.BB1.AVERage.COUNt = Var
Var = SCPI.SENSe.BB1.AVERage.COUNt
```

Related Objects

SCPI.SENSe.BB(Ch).AVERage.CLEar

SCPI.SENSe.BB(Ch).AVERage.STATe

Equivalent Key

BB Menu: **Average** > **Avg Factor**

Equivalent SCPI Command

Syntax

:SENSe:BB[1-1]:AVERage:COUNt {1 ~ 999}

:SENSe:BB[1-1]:AVERage:COUNt?

Query Response

{1 ~ 999} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:BB1:AVERage:COUNt 1 "
20 OUTPUT 717;":SENSe:BB1:AVERage:COUNt ?"
30 ENTER 717;A
```

SCPI.SENSe.BB(Ch).AVERage.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.BB(Ch).AVERage.STATe = *Value*

Value = SCPI.SENSe.BB(Ch).AVERage.STATe

Description

This command sets/gets average ON/OFF, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Average (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.SENSe.BB1.AVERage.STATe = Var
Var = SCPI.SENSe.BB1.AVERage.STATe
```

Related Objects

SCPI.SENSe.BB(Ch).AVERage.CLEar

SCPI.SENSe.BB(Ch).AVERage.COUNt

Equivalent Key

BB Menu: **Average** > **Averaging**

Equivalent SCPI Command

Syntax

:SENSe:BB[1-1]:AVERage:STATe {ON|OFF|1|0}

:SENSe:BB[1-1]:AVERage:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:BB1:AVERage:STATe 1"
20 OUTPUT 717;":SENSe:BB1:AVERage:STATe ?"
30 ENTER 717;A
```

SCPI.SENSe.BB(Ch).COFRequency

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.BB(Ch).COFRequency = *Value*

Value = SCPI.SENSe.BB(Ch).COFRequency

Description

This command sets/gets DC Block cut off frequency, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	DC Block cut off frequency
Data Type	Character string type (String)
Range	LOW|HIGH
Preset Value	LOW
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "LOW"
SCPI.SENSe.BB1.COFRequency = Var
Var = SCPI.SENSe.BB1.COFRequency
```

Equivalent Key

BB Menu: **Setup** > **Cutoff Frequency**

Equivalent SCPI Command

Syntax

:SENSe:BB[1-1]:COFRequency {LOW|HIGH}

:SENSe:BB[1-1]:COFRequency?

Query Response

{LOW|HIGH} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:BB1:COFRequency LOW"
20 OUTPUT 717;":SENSe:BB1:COFRequency ?"
30 ENTER 717;A$
```

SCPI.SENSe.BB(Ch).CORRelation.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.BB(Ch).CORRelation.COUNt = *Value*

Value = SCPI.SENSe.BB(Ch).CORRelation.COUNt

Description

This command sets/gets correlation count, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Correlation count
Data Type	Long integer type (Long)
Range	1 ~ 10000
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.SENSe.BB1.CORRelation.COUNt = Var
Var = SCPI.SENSe.BB1.CORRelation.COUNt
```

Equivalent Key

BB Menu: **Average** > **Correlation**

Equivalent SCPI Command

Syntax

:SENSe:BB[1-1]:CORRelation:COUNt {1 ~ 10000}

:SENSe:BB[1-1]:CORRelation:COUNt?

Query Response

{1 ~ 10000} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:BB1:CORRelation:COUNt 1 "
20 OUTPUT 717;":SENSe:BB1:CORRelation:COUNt ?"
30 ENTER 717;A
```

SCPI.SENSe.BB(Ch).DISCharge

Object Type

Method (**Write Only**)

Syntax

SCPI.SENSe.BB(Ch).DISCharge

Description

This command sets Discharge DC Block capacitor, for the selected channel *Ch*.

Examples

SCPI.SENSe.BB1.DISCharge

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SENSe:BB[1-1]:DISCharge

Example of use

```
10 OUTPUT 717;":SENSe:BB1:DISCharge"
```

SCPI.SENSe.BB(Ch).FREQuency.STARt

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.BB(Ch).FREQuency.STARt = *Value*

Value = SCPI.SENSe.BB(Ch).FREQuency.STARt

Description

This command sets/gets start frequency, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Start frequency
Data Type	Double precision floating point type (Double)
Range	1|10|100|1k
Preset Value	1k
Unit	Hz
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SENSe.BB1.FREQuency.STARt = Var
Var = SCPI.SENSe.BB1.FREQuency.STARt
```

Related Objects

SCPI.SENSe.BB(Ch).FREQuency.STOP

Equivalent Key

BB Menu: **Start** > **1Hz**

BB Menu: **Start** > **10Hz**

BB Menu: **Start** > **100Hz**

BB Menu: **Start** > **1kHz**

BB Menu: **Marker To** > **Marker** > **Start**

Equivalent SCPI Command

Syntax

:SENSe:BB[1-1]:FREQuency:STARt {1|10|100|1k}

:SENSe:BB[1-1]:FREQuency:STARt?

Query Response

{1|10|100|1k} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:BB1:FREQuency:STARt 0"
20 OUTPUT 717;":SENSe:BB1:FREQuency:STARt ?"
30 ENTER 717;A
```

SCPI.SENSe.BB(Ch).FREQuency.STOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.BB(Ch).FREQuency.STOP = *Value*

Value = SCPI.SENSe.BB(Ch).FREQuency.STOP

Description

This command sets/gets stop frequency, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Stop frequency
Data Type	Double precision floating point type (Double)
Range	100k|1M|5M|10M|20M|40M|100M
Preset Value	10M
Unit	Hz
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 100000
SCPI.SENSe.BB1.FREQuency.STOP = Var
Var = SCPI.SENSe.BB1.FREQuency.STOP
```

Related Objects

SCPI.SENSe.BB(Ch).FREQuency.STARt

Equivalent Key

BB Menu: **Stop** > **100kHz**

BB Menu: **Stop** > **1MHz**

BB Menu: **Stop** > **5MHz**

BB Menu: **Stop** > **10MHz**

BB Menu: **Stop** > **20MHz**

BB Menu: **Stop** > **40MHz**

BB Menu: **Stop** > **100MHz**

BB Menu: **Marker To** > **Marker** > **Stop**

Equivalent SCPI Command

Syntax

:SENSe:BB[1-1]:FREQuency:STOP {100k|1M|5M|10M|20M|40M|100M}

:SENSe:BB[1-1]:FREQuency:STOP?

Query Response

{100k|1M|5M|10M|20M|40M|100M} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:BB1:FREQuency:STOP 100000"
20 OUTPUT 717;":SENSe:BB1:FREQuency:STOP ?"
30 ENTER 717;A
```

SCPI.SENSe.BB(Ch).IFGain

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.BB(Ch).IFGain = *Value*

Value = SCPI.SENSe.BB(Ch).IFGain

Description

This command sets/gets IF IFGain 10dB Step, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	IF IFGain 10dB Step
Data Type	Double precision floating point type (Double)
Range	0 ~ 50
Preset Value	20
Unit	dB
Resolution	10
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SENSe.BB1.IFGain = Var
Var = SCPI.SENSe.BB1.IFGain
```

Equivalent Key

BB Menu: **Setup** > **IF Gain**

Equivalent SCPI Command

Syntax

:SENSe:BB[1-1]:IFGain {0 ~ 50}

:SENSe:BB[1-1]:IFGain?

Query Response

{0 ~ 50} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:BB1:IFGain 0"
20 OUTPUT 717;":SENSe:BB1:IFGain ?"
30 ENTER 717;A
```

SCPI.SENSe.BB(Ch).SEGTable.MEASurement.QUALity

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.BB(Ch).SEGTable.MEASurement.QUALity = *Value*

Value = SCPI.SENSe.BB(Ch).SEGTable.MEASurement.QUALity

Description

This command sets/gets specifies which segment table to use, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Specifies which segment table to use
Data Type	Character string type (String)
Range	NORMal|FAST
Preset Value	NORMal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NORMal"
SCPI.SENSe.BB1.SEGTable.MEASurement.QUALity = Var
Var = SCPI.SENSe.BB1.SEGTable.MEASurement.QUALity
```

Equivalent Key

BB Menu: **Setup** > **Measurement Quality**

Equivalent SCPI Command

Syntax

:SENSe:BB[1-1]:SEGTable[:MEASurement][:QUALity] {NORMal|FAST}

:SENSe:BB[1-1]:SEGTable[:MEASurement][:QUALity]?

Query Response

{NORMal|FAST} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:BB1:SEGTable[:MEASurement][:QUALity] NORMal"
20 OUTPUT 717;":SENSe:BB1:SEGTable[:MEASurement][:QUALity] ?"
30 ENTER 717;A$
```

SCPI.SENSe.BB(Ch).SWEep.POINts

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.SENSe.BB(Ch).SWEep.POINts

Description

This command gets number of points, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Number of points
Data Type	Long integer type (Long)
Range	2 ~ 1601
Preset Value	517
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 2
Var = SCPI.SENSe.BB1.SWEep.POINts
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SENSe:BB[1-1]:SWEep:POINts?

Query Response

{2 ~ 1601} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:BB1:SWEep:POINts ?"
20 ENTER 717;A
```

SCPI.SENSe.CORRection.POWer.DATA

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.CORRection.POWer.DATA = *Value*

Value = SCPI.SENSe.CORRection.POWer.DATA

Description

This command sets/gets correction table (list of). freq must be incremental order..

Variable

Parameter	*Value*
Description	Correction table
Data Type	Variant type Array (Range)
Range	1...2002
Preset Value	-
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-224, "Illegal parameter value"

Examples

```
Dim Var(1 to 2002) as Variant
SCPI.SENSe.CORRection.POWer.DATA = Var
Var = SCPI.SENSe.CORRection.POWer.DATA
```

Related Objects

SCPI.SENSe.CORRection.POWer.STATe

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SENSe:CORRection:POWer:DATA {1...2002}

:SENSe:CORRection:POWer:DATA?

Query Response

{1...2002} <newline><^END>

Example of use

```
10 Dim A(1:2002)
20 OUTPUT 717;":SENSe:CORRection:POWer:DATA 1,2002"
30 OUTPUT 717;":SENSe:CORRection:POWer:DATA ?"
40 ENTER 717;A(*)
```

SCPI.SENSe.CORRection.POWer.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.CORRection.POWer.STATe = *Value*

Value = SCPI.SENSe.CORRection.POWer.STATe

Description

This command sets/gets power correction On/Off.

Variable

Parameter	*Value*
Description	Power correction (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.SENSe.CORRection.POWer.STATe = Var
Var = SCPI.SENSe.CORRection.POWer.STATe
```

Related Objects

SCPI.SENSe.CORRection.POWer.DATA

Equivalent Key

PN Menu: **System** > **Instrument Setup** > **Correction** > **Power Correction**

SP Menu: **System** > **Instrument Setup** > **Correction** > **Power Correction**

FP Menu: **System** > **Instrument Setup** > **Correction** > **Power Correction**

TR Menu: **System** > **Instrument Setup** > **Correction** > **Power Correction**

AM Menu: **System** > **Instrument Setup** > **Correction** > **Power Correction**

BB Menu: **System** > **Instrument Setup** > **Correction** > **Power Correction**

USER Menu: **System** > **Instrument Setup** > **Correction** > **Power Correction**

Equivalent SCPI Command

Syntax

:SENSe:CORRection:POWer:STATe {ON|OFF|1|0}

:SENSe:CORRection:POWer:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:CORRection:POWer:STATe 1"
20 OUTPUT 717;":SENSe:CORRection:POWer:STATe ?"
30 ENTER 717;A
```

SCPI.SENSe.DCONverter.IDN

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.SENSe.DCONverter.IDN

Description

This command gets downconverter model information.

Variable

Parameter	*Value*
Description	Downconverter model information
Data Type	Character string type (String)
Range	254 chars
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as String
Var = SCPI.SENSe.DCONverter.IDN
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SENSe:DCONverter:IDN?

Query Response

{String}<newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:DCONverter:IDN ?"
20 ENTER 717;A$
```

SCPI.SENSe.DCONverter.INPut

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.DCONverter.INPut = *Value*

Value = SCPI.SENSe.DCONverter.INPut

Description

This command sets/gets RF Input port.

Variable

Parameter	*Value*
Description	RF Input port
Data Type	Character string type (String)
Range	DCONverter|DIRect
Preset Value	DCONverter
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "DCONverter"
SCPI.SENSe.DCONverter.INPut = Var
Var = SCPI.SENSe.DCONverter.INPut
```

Equivalent Key

PN Menu: **Input Port** > **Downconverter** > **RF Input**

SP Menu: **Input Port** > **Downconverter** > **RF Input**

FP Menu: **Input Port** > **Downconverter** > **RF Input**

TR Menu: **Input Port** > **Downconverter** > **RF Input**

AM Menu: **Input Port** > **Downconverter** > **RF Input**

BB Menu: **Input Port** > **Downconverter** > **RF Input**

USER Menu: **Input Port** > **Downconverter** > **RF Input**

Equivalent SCPI Command

Syntax

:SENSe:DCONverter:INPut {DCONverter|DIRect}

:SENSe:DCONverter:INPut?

Query Response

{DCONverter|DIRect} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:DCONverter:INPut DCONverter"
20 OUTPUT 717;":SENSe:DCONverter:INPut ?"
30 ENTER 717;A$
```

SCPI.SENSe.DCONverter.MANual.CALCulate.LO_Q harmonic, in_freq, lo1, lo2

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.SENSe.DCONverter.MANual.CALCulate.LO_Q harmonic, in_freq, lo1, lo2

Description

This command gets Query recommended LO1, LO2 frequency for the specified input frequency and harmonic number.

Variable

Parameter	*Value*
Description	LO1, LO2 frequency for the specified input frequency and harmonic number
Data Type	Variant type Array (Range)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No

Error	-

Parameter	*harmonic*
Description	Harmonic number
Data Type	Long integer type (Long)
Range	1 ~ 34
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*input frequency*
Description	Input frequency
Data Type	Double precision floating point type (Double)
Range	3G ~ 327G

Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*LO1*
Description	LO 1
Data Type	Character string type (String)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*LO2*
Description	LO 2
Data Type	Character string type (String)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
Var = SCPI.SENSe.DCONverter.MANual.CALCulate.LO_Q harmonic, in_freq, lo1, lo2
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SENSe:DCONverter:MANual:CALCulate:LO?

Query Response

{1 ~ 34 3G ~ 327G - -} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:DCONverter:MANual:CALCulate:LO ?"
30 ENTER 717;A$
```

SCPI.SENSe.DCONverter.MANual.IFDelta

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.DCONverter.MANual.IFDelta = *Value*

Value = SCPI.SENSe.DCONverter.MANual.IFDelta

Description

This command sets/gets down converter delta IF frequency (IF2 - IF1).

Variable

Parameter	*Value*
Description	Down converter delta IF frequency (IF2 - IF1)
Data Type	Double precision floating point type (Double)
Range	-3G ~ 3G
Preset Value	0
Unit	Hz
Resolution	1
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -1000000000
SCPI.SENSe.DCONverter.MANual.IFDelta = Var
Var = SCPI.SENSe.DCONverter.MANual.IFDelta
```

Related Objects

SCPI.SENSe.DCONverter.MANual.IFGain(1-2)

Equivalent Key

PN Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **ÄIF = IF2 - IF1**

SP Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **ÄIF = IF2 - IF1**

FP Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **ÄIF = IF2 - IF1**

TR Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **ÄIF = IF2 - IF1**

AM Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **ÄIF = IF2 - IF1**

BB Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **ÄIF = IF2 - IF1**

USER Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **ÄIF = IF2 - IF1**

Equivalent SCPI Command

Syntax

:SENSe:DCONverter:MANual:IFDelta {-3G ~ 3G}

:SENSe:DCONverter:MANual:IFDelta?

Query Response

{-3G ~ 3G} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:DCONverter:MANual:IFDelta -1000000000"
20 OUTPUT 717;":SENSe:DCONverter:MANual:IFDelta ?"
30 ENTER 717;A
```

SCPI.SENSe.DCONverter.MANual.IFGain(1-2)

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.DCONverter.MANual.IFGain(1-2) = *Value*

Value = SCPI.SENSe.DCONverter.MANual.IFGain(1-2)

Description

This command sets/gets down converter IF Gain.

Variable

Parameter	*Value*
Description	Down converter IF Gain
Data Type	Double precision floating point type (Double)
Range	0 ~ 35
Preset Value	0
Unit	dB
Resolution	5
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SENSe.DCONverter.MANual.IFGain2 = Var
Var = SCPI.SENSe.DCONverter.MANual.IFGain2
```

Related Objects

SCPI.SENSe.DCONverter.MANual.IFDelta

Equivalent Key

PN Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **IF Gain 1**

SP Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **IF Gain 1**

FP Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **IF Gain 1**

TR Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **IF Gain 1**

AM Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **IF Gain 1**

BB Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **IF Gain 1**

USER Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **IF Gain 1**

Equivalent SCPI Command

Syntax

:SENSe:DCONverter:MANual:IFGain[1-2] {0 ~ 35}

:SENSe:DCONverter:MANual:IFGain[1-2]?

Query Response

{0 ~ 35} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:DCONverter:MANual:IFGain2 0"
20 OUTPUT 717;":SENSe:DCONverter:MANual:IFGain2 ?"
30 ENTER 717;A
```

SCPI.SENSe.DCONverter.MANual.LO(1-2).FREQuency

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.DCONverter.MANual.LO(1-2).FREQuency = *Value*

Value = SCPI.SENSe.DCONverter.MANual.LO(1-2).FREQuency

Description

This command sets/gets down converter LO Frequency.

Variable

Parameter	*Value*
Description	Down converter LO Frequency
Data Type	Double precision floating point type (Double)
Range	2.975G ~ 10.025G
Preset Value	2.975G
Unit	Hz
Resolution	100m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 2975000000
SCPI.SENSe.DCONverter.MANual.LO2.FREQuency = Var
Var = SCPI.SENSe.DCONverter.MANual.LO2.FREQuency
```

Related Objects

SCPI.SENSe.DCONverter.MANual.LO(1-2).LEVel

Equivalent Key

PN Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **LO1 Frequency**

SP Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **LO1 Frequency**

FP Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **LO1 Frequency**

TR Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **LO1 Frequency**

AM Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **LO1 Frequency**

BB Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **LO1 Frequency**

USER Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **LO1 Frequency**

Equivalent SCPI Command

Syntax

:SENSe:DCONverter:MANual:LO[1-2]:FREQuency {2.975G ~ 10.025G}

:SENSe:DCONverter:MANual:LO[1-2]:FREQuency?

Query Response

{2.975G ~ 10.025G} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:DCONverter:MANual:LO2:FREQuency 2975000000"
20 OUTPUT 717;":SENSe:DCONverter:MANual:LO2:FREQuency ?"
30 ENTER 717;A
```

SCPI.SENSe.DCONverter.MANual.LO(1-2).LEVel

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.DCONverter.MANual.LO(1-2).LEVel = *Value*

Value = SCPI.SENSe.DCONverter.MANual.LO(1-2).LEVel

Description

This command sets/gets down converter LO Output Level.

Variable

Parameter	*Value*
Description	Down converter LO Output Level
Data Type	Double precision floating point type (Double)
Range	10 ~ 16
Preset Value	10
Unit	dBm
Resolution	500m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SENSe.DCONverter.MANual.LO2.LEVel = Var
Var = SCPI.SENSe.DCONverter.MANual.LO2.LEVel
```

Related Objects

SCPI.SENSe.DCONverter.MANual.LO(1-2).FREQuency

Equivalent Key

PN Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **LO1 Level**

SP Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **LO1 Level**

FP Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **LO1 Level**

TR Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **LO1 Level**

AM Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **LO1 Level**

BB Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **LO1 Level**

USER Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **LO1 Level**

Equivalent SCPI Command

Syntax

:SENSe:DCONverter:MANual:LO[1-2]:LEVel {10 ~ 16}

:SENSe:DCONverter:MANual:LO[1-2]:LEVel?

Query Response

{10 ~ 16} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:DCONverter:MANual:LO2:LEVel 0"
20 OUTPUT 717;":SENSe:DCONverter:MANual:LO2:LEVel ?"
30 ENTER 717;A
```

SCPI.SENSe.DCONverter.MANual.MEXTernal(1-2).BIAS.CURRent

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.DCONverter.MANual.MEXTernal(1-2).BIAS.CURRent = *Value*

Value = SCPI.SENSe.DCONverter.MANual.MEXTernal(1-2).BIAS.CURRent

Description

This command sets/gets external mixer bias current.

Variable

Parameter	*Value*
Description	External mixer bias current
Data Type	Double precision floating point type (Double)
Range	-10m ~ 10m
Preset Value	0
Unit	A
Resolution	10u
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -0.01
SCPI.SENSe.DCONverter.MANual.MEXTernal2.BIAS.CURRent = Var
Var = SCPI.SENSe.DCONverter.MANual.MEXTernal2.BIAS.CURRent
```

Related Objects

SCPI.SENSe.DCONverter.MANual.MEXTernal(1-2).BIAS.STATe

Equivalent Key

PN Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **Current**

SP Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **Current**

FP Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **Current**

TR Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **Current**

AM Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **Current**

BB Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **Current**

USER Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **Current**

Equivalent SCPI Command

Syntax

:SENSe:DCONverter:MANual:MEXTernal[1-2]:BIAS:CURRent {-10m ~ 10m}

:SENSe:DCONverter:MANual:MEXTernal[1-2]:BIAS:CURRent?

Query Response

{-10m ~ 10m} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:DCONverter:MANual:MEXTernal2:BIAS:CURRent -0.01"
20 OUTPUT 717;":SENSe:DCONverter:MANual:MEXTernal2:BIAS:CURRent ?"
30 ENTER 717;A
```

SCPI.SENSe.DCONverter.MANual.MEXTernal(1-2).BIAS.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.DCONverter.MANual.MEXTernal(1-2).BIAS.STATe = *Value*

Value = SCPI.SENSe.DCONverter.MANual.MEXTernal(1-2).BIAS.STATe

Description

This command sets/gets external mixer bias state.

Variable

Parameter	*Value*
Description	External mixer bias state
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.SENSe.DCONverter.MANual.MEXTernal2.BIAS.STATe = Var
Var = SCPI.SENSe.DCONverter.MANual.MEXTernal2.BIAS.STATe
```

Related Objects

SCPI.SENSe.DCONverter.MANual.MEXTernal(1-2).BIAS.CURRent

Equivalent Key

PN Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **Mixer 1 Bias**

SP Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **Mixer 1 Bias**

FP Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **Mixer 1 Bias**

TR Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **Mixer 1 Bias**

AM Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **Mixer 1 Bias**

BB Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **Mixer 1 Bias**

USER Menu: **System** > **Instrument Setup** > **Downconverter Manual Setup** > **Mixer 1 Bias**

Equivalent SCPI Command

Syntax

:SENSe:DCONverter:MANual:MEXTernal[1-2]:BIAS:STATe {ON|OFF|1|0}

:SENSe:DCONverter:MANual:MEXTernal[1-2]:BIAS:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:DCONverter:MANual:MEXTernal2:BIAS:STATe 1"
20 OUTPUT 717;":SENSe:DCONverter:MANual:MEXTernal2:BIAS:STATe ?"
30 ENTER 717;A
```

SCPI.SENSe.DCONverter.MEXTernal

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.DCONverter.MEXTernal = *Value*

Value = SCPI.SENSe.DCONverter.MEXTernal

Description

This command sets/gets external mixer selection.

Variable

Parameter	*Value*
Description	External mixer selection
Data Type	Character string type (String)
Range	OFF|ON
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "OFF"
SCPI.SENSe.DCONverter.MEXTernal = Var
Var = SCPI.SENSe.DCONverter.MEXTernal
```

Equivalent Key

PN Menu: **Input Port** > **Downconverter** > **External Mixer**

SP Menu: **Input Port** > **Downconverter** > **External Mixer**

FP Menu: **Input Port** > **Downconverter** > **External Mixer**

TR Menu: **Input Port** > **Downconverter** > **External Mixer**

AM Menu: **Input Port** > **Downconverter** > **External Mixer**

BB Menu: **Input Port** > **Downconverter** > **External Mixer**

USER Menu: **Input Port** > **Downconverter** > **External Mixer**

Equivalent SCPI Command

Syntax

:SENSe:DCONverter:MEXTernal {OFF|ON}

:SENSe:DCONverter:MEXTernal?

Query Response

{OFF|ON} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:DCONverter:MEXTernal OFF"
20 OUTPUT 717;":SENSe:DCONverter:MEXTernal ?"
30 ENTER 717;A$
```

SCPI.SENSe.DCONverter.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.DCONverter.STATe = *Value*

Value = SCPI.SENSe.DCONverter.STATe

Description

This command sets/gets down converter usage.

Variable

Parameter	*Value*
Description	Down converter usage
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.SENSe.DCONverter.STATe = Var
Var = SCPI.SENSe.DCONverter.STATe
```

Equivalent Key

PN Menu: **Input Port** > **Downconverter** > **Downconverter**

SP Menu: **Input Port** > **Downconverter** > **Downconverter**

FP Menu: **Input Port** > **Downconverter** > **Downconverter**

TR Menu: **Input Port** > **Downconverter** > **Downconverter**

AM Menu: **Input Port** > **Downconverter** > **Downconverter**

BB Menu: **Input Port** > **Downconverter** > **Downconverter**

USER Menu: **Input Port** > **Downconverter** > **Downconverter**

Equivalent SCPI Command

Syntax

:SENSe:DCONverter[:STATe] {ON|OFF|1|0}

:SENSe:DCONverter[:STATe]?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:DCONverter[:STATe] 1"
20 OUTPUT 717;":SENSe:DCONverter[:STATe] ?"
30 ENTER 717;A
```

SCPI.SENSe.FP(Ch).AVERage.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.SENSe.FP(Ch).AVERage.CLEar

Description

This command clears average, for the selected channel *Ch*.

Examples

SCPI.SENSe.FP1.AVERage.CLEar

Related Objects

SCPI.SENSe.FP(Ch).AVERage.COUNt

SCPI.SENSe.FP(Ch).AVERage.STATe

Equivalent Key

FP Menu: **Average** > **Averaging Restart**

Equivalent SCPI Command

Syntax

:SENSe:FP[1-1]:AVERage:CLEar

Example of use

```
10 OUTPUT 717;":SENSe:FP1:AVERage:CLEar"
```

SCPI.SENSe.FP(Ch).AVERage.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.FP(Ch).AVERage.COUNt = *Value*

Value = SCPI.SENSe.FP(Ch).AVERage.COUNt

Description

This command sets/gets average count, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Average count
Data Type	Long integer type (Long)
Range	1 ~ 999
Preset Value	16
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.SENSe.FP1.AVERage.COUNt = Var
Var = SCPI.SENSe.FP1.AVERage.COUNt
```

Related Objects

SCPI.SENSe.FP(Ch).AVERage.CLEar

SCPI.SENSe.FP(Ch).AVERage.STATe

Equivalent Key

FP Menu: **Average** > **Avg Factor**

Equivalent SCPI Command

Syntax

:SENSe:FP[1-1]:AVERage:COUNt {1 ~ 999}

:SENSe:FP[1-1]:AVERage:COUNt?

Query Response

{1 ~ 999} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:FP1:AVERage:COUNt 1 "
20 OUTPUT 717;":SENSe:FP1:AVERage:COUNt ?"
30 ENTER 717;A
```

SCPI.SENSe.FP(Ch).AVERage.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.FP(Ch).AVERage.STATe = *Value*

Value = SCPI.SENSe.FP(Ch).AVERage.STATe

Description

This command sets/gets average ON/OFF, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Average (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.SENSe.FP1.AVERage.STATe = Var
Var = SCPI.SENSe.FP1.AVERage.STATe
```

Related Objects

SCPI.SENSe.FP(Ch).AVERage.CLEar

SCPI.SENSe.FP(Ch).AVERage.COUNt

Equivalent Key

FP Menu: **Average** > **Averaging**

Equivalent SCPI Command

Syntax

:SENSe:FP[1-1]:AVERage:STATe {ON|OFF|1|0}

:SENSe:FP[1-1]:AVERage:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:FP1:AVERage:STATe 1"
20 OUTPUT 717;":SENSe:FP1:AVERage:STATe ?"
30 ENTER 717;A
```

SCPI.SENSe.FP(Ch).DCONverter.FREQuency

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.FP(Ch).DCONverter.FREQuency = *Value*

Value = SCPI.SENSe.FP(Ch).DCONverter.FREQuency

Description

This command sets/gets downconverter input frequency (carrier nominal frequency), for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Downconverter input frequency (carrier nominal frequency)
Data Type	Double precision floating point type (Double)
Range	3G ~ 26.5G
Preset Value	3G
Unit	Hz
Resolution	100m
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
Var= 3000000000
SCPI.SENSe.FP1.DCONverter.FREQuency = Var
Var = SCPI.SENSe.FP1.DCONverter.FREQuency
```

Equivalent Key

FP Menu: **Setup** > **Nominal Frequency**

Equivalent SCPI Command

Syntax

:SENSe:FP[1-1]:DCONverter:FREQuency {3G ~ 26.5G}

:SENSe:FP[1-1]:DCONverter:FREQuency?

Query Response

{3G ~ 26.5G} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:FP1:DCONverter:FREQuency 3000000000"
20 OUTPUT 717;":SENSe:FP1:DCONverter:FREQuency ?"
30 ENTER 717;A
```

SCPI.SENSe.FP(Ch).DCONverter.SSEarch.EXECute

Object Type

Method (**Write Only**)

Syntax

SCPI.SENSe.FP(Ch).DCONverter.SSEarch.EXECute

Description

This command searches carrier signal, for the selected channel *Ch*.

Examples

SCPI.SENSe.FP1.DCONverter.SSEarch.EXECute

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SENSe:FP[1-1]:DCONverter:SSEarch:EXECute

Example of use

10 OUTPUT 717;":SENSe:FP1:DCONverter:SSEarch:EXECute"

SCPI.SENSe.FP(Ch).FBANd

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.FP(Ch).FBANd = *Value*

Value = SCPI.SENSe.FP(Ch).FBANd

Description

This command sets/gets frequency band, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Frequency band
Data Type	Character string type (String)
Range	LOW|HIGH|BAND3|BAND4
Preset Value	HIGH
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

When the E5052B is used stand-alone, or with the downconverter turned off

Parameter	*<string>*
LOW	Set frequency band to '10M - 1.5GHz'
HIGH(*Preset* value)	Set frequency band to '250M - 7GHz'

When the downconverter is turned on and with the *RF* input is set to 'E5052B Direct'

Parameter	*<string>*
LOW	Set frequency band to '10M - 1.5GHz'
HIGH(Preset value)	Set frequency band to '250M - 3GHz'

When the downconverter is turned on and with the RF input is set to 'Downconverter'

Parameter	*<string>*
BAND3 (Preset value)	Set frequency band to '3G - 10GHz'
BAND4	Set frequency band to '9G - 26.5GHz'

Examples

```
Dim Var as String
Var= "LOW"
SCPI.SENSe.FP1.FBANd = Var
Var = SCPI.SENSe.FP1.FBANd
```

Equivalent Key

FP Menu: **Setup** > **Frequency Band**

Equivalent SCPI Command

Syntax

:SENSe:FP[1-1]:FBANd {LOW|HIGH|BAND3|BAND4}

:SENSe:FP[1-1]:FBANd?

Query Response

{LOW|HIGH|BAND3|BAND4} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:FP1:FBANd LOW"
20 OUTPUT 717;":SENSe:FP1:FBANd ?"
30 ENTER 717;A$
```

SCPI.SENSe.FP(Ch).FREQuency.RESolution

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.FP(Ch).FREQuency.RESolution = *Value*

Value = SCPI.SENSe.FP(Ch).FREQuency.RESolution

Description

This command sets/gets frequency resolution 1KHz or 64KHz, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Frequency Resolution 1KHz or 64KHz
Data Type	Character string type (String)
Range	NARRow|MIDDle|WIDE
Preset Value	WIDE
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NARRow"
SCPI.SENSe.FP1.FREQuency.RESolution = Var
Var = SCPI.SENSe.FP1.FREQuency.RESolution
```

Equivalent Key

FP Menu: **Setup** > **Freq Resolution**

Equivalent SCPI Command

Syntax

:SENSe:FP[1-1]:FREQuency:RESolution {NARRow|MIDDle|WIDE}

:SENSe:FP[1-1]:FREQuency:RESolution?

Query Response

{NARRow|MIDDle|WIDE} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:FP1:FREQuency:RESolution NARRow"
20 OUTPUT 717;":SENSe:FP1:FREQuency:RESolution ?"
30 ENTER 717;A$
```

SCPI.SENSe.FP(Ch).POWer.INPut.LEVel.MAXimum

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.FP(Ch).POWer.INPut.LEVel.MAXimum = *Value*

Value = SCPI.SENSe.FP(Ch).POWer.INPut.LEVel.MAXimum

Description

This command sets/gets max input level, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Max Input Level
Data Type	Double precision floating point type (Double)
Range	-45 ~ 30
Preset Value	0
Unit	dBm
Resolution	100m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SENSe.FP1.POWer.INPut.LEVel.MAXimum = Var
Var = SCPI.SENSe.FP1.POWer.INPut.LEVel.MAXimum
```

Equivalent Key

FP Menu: **Setup** > **Max Input Level**

Equivalent SCPI Command

Syntax

:SENSe:FP[1-1]:POWer:INPut:LEVel:MAXimum {-45 ~ 30}

:SENSe:FP[1-1]:POWer:INPut:LEVel:MAXimum?

Query Response

{-45 ~ 30} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:FP1:POWer:INPut:LEVel:MAXimum 0"
20 OUTPUT 717;":SENSe:FP1:POWer:INPut:LEVel:MAXimum ?"
30 ENTER 717;A
```

SCPI.SENSe.FP(Ch).SWEep.DWELl

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.FP(Ch).SWEep.DWELl = *Value*

Value = SCPI.SENSe.FP(Ch).SWEep.DWELl

Description

This command sets/gets point delay, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Point delay
Data Type	Double precision floating point type (Double)
Range	0 ~ 1
Preset Value	0
Unit	s
Resolution	100u
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SENSe.FP1.SWEep.DWELl = Var
Var = SCPI.SENSe.FP1.SWEep.DWELl
```

Equivalent Key

FP Menu: **Setup** > **Point Delay**

Equivalent SCPI Command

Syntax

:SENSe:FP[1-1]:SWEep:DWELl {0 ~ 1}

:SENSe:FP[1-1]:SWEep:DWELl?

Query Response

{0 ~ 1} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:FP1:SWEep:DWELl 0"
20 OUTPUT 717;":SENSe:FP1:SWEep:DWELl ?"
30 ENTER 717;A
```

SCPI.SENSe.FP(Ch).SWEep.TIME.DATA

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.SENSe.FP(Ch).SWEep.TIME.DATA

Description

This command gets sweep time resolution (1mSec), for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Sweep time Resolution (1mSec)
Data Type	Double precision floating point type (Double)
Range	-
Preset Value	-
Unit	s
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var = SCPI.SENSe.FP1.SWEep.TIME.DATA
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SENSe:FP[1-1]:SWEep:TIME:DATA?

Query Response

{Double} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:FP1:SWEep:TIME:DATA ?"
20 ENTER 717;A
```

SCPI.SENSe.PN(Ch).ASET

Object Type

Method (**Write Only**)

Syntax

SCPI.SENSe.PN(Ch).ASET

Description

This command sets Attn, Freq Band and IF Gain Auto Set, for the selected channel *Ch*.

Examples

SCPI.SENSe.PN1.ASET

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SENSe:PN[1-1]:ASET

Example of use

10 OUTPUT 717;":SENSe:PN1:ASET"

SCPI.SENSe.PN(Ch).AVERage.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.SENSe.PN(Ch).AVERage.CLEar

Description

This command clears average, for the selected channel *Ch*.

Examples

SCPI.SENSe.PN1.AVERage.CLEar

Related Objects

SCPI.SENSe.PN(Ch).AVERage.COUNt

SCPI.SENSe.PN(Ch).AVERage.STATe

Equivalent Key

PN Menu: **Average** > **Averaging Restart**

Equivalent SCPI Command

Syntax

:SENSe:PN[1-1]:AVERage:CLEar

Example of use

```
10 OUTPUT 717;":SENSe:PN1:AVERage:CLEar"
```

SCPI.SENSe.PN(Ch).AVERage.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PN(Ch).AVERage.COUNt = *Value*

Value = SCPI.SENSe.PN(Ch).AVERage.COUNt

Description

This command sets/gets average count, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Average count
Data Type	Long integer type (Long)
Range	1 ~ 999
Preset Value	16
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.SENSe.PN1.AVERage.COUNt = Var
Var = SCPI.SENSe.PN1.AVERage.COUNt
```

Related Objects

SCPI.SENSe.PN(Ch).AVERage.CLEar

SCPI.SENSe.PN(Ch).AVERage.STATe

Equivalent Key

PN Menu: **Average** > **Avg Factor**

Equivalent SCPI Command

Syntax

:SENSe:PN[1-1]:AVERage:COUNt {1 ~ 999}

:SENSe:PN[1-1]:AVERage:COUNt?

Query Response

{1 ~ 999} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PN1:AVERage:COUNt 1 "
20 OUTPUT 717;":SENSe:PN1:AVERage:COUNt ?"
30 ENTER 717;A
```

SCPI.SENSe.PN(Ch).AVERage.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PN(Ch).AVERage.STATe = *Value*

Value = SCPI.SENSe.PN(Ch).AVERage.STATe

Description

This command sets/gets average ON/OFF, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Average (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.SENSe.PN1.AVERage.STATe = Var
Var = SCPI.SENSe.PN1.AVERage.STATe
```

Related Objects

SCPI.SENSe.PN(Ch).AVERage.CLEar

SCPI.SENSe.PN(Ch).AVERage.COUNt

Equivalent Key

PN Menu: **Average** > **Averaging**

Equivalent SCPI Command

Syntax

:SENSe:PN[1-1]:AVERage:STATe {ON|OFF|1|0}

:SENSe:PN[1-1]:AVERage:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PN1:AVERage:STATe 1"
20 OUTPUT 717;":SENSe:PN1:AVERage:STATe ?"
30 ENTER 717;A
```

SCPI.SENSe.PN(Ch).CORRelation.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PN(Ch).CORRelation.COUNt = *Value*

Value = SCPI.SENSe.PN(Ch).CORRelation.COUNt

Description

This command sets/gets correlation count, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Correlation count
Data Type	Long integer type (Long)
Range	1 ~ 10000
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.SENSe.PN1.CORRelation.COUNt = Var
Var = SCPI.SENSe.PN1.CORRelation.COUNt
```

Equivalent Key

PN Menu: **Average** > **Correlation**

Equivalent SCPI Command

Syntax

:SENSe:PN[1-1]:CORRelation:COUNt {1 ~ 10000}

:SENSe:PN[1-1]:CORRelation:COUNt?

Query Response

{1 ~ 10000} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PN1:CORRelation:COUNt 1 "
20 OUTPUT 717;":SENSe:PN1:CORRelation:COUNt ?"
30 ENTER 717;A
```

SCPI.SENSe.PN(Ch).CRANge

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PN(Ch).CRANge = *Value*

Value = SCPI.SENSe.PN(Ch).CRANge

Description

This command sets/gets capture range, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Capture range
Data Type	Character string type (String)
Range	NORMal|WIDE
Preset Value	NORMal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NORMal"
SCPI.SENSe.PN1.CRANge = Var
Var = SCPI.SENSe.PN1.CRANge
```

Equivalent Key

PN Menu: **Setup** > **Capture Range**

Equivalent SCPI Command

Syntax

:SENSe:PN[1-1]:CRANge {NORMal|WIDE}

:SENSe:PN[1-1]:CRANge?

Query Response

{NORMal|WIDE} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PN1:CRANge NORMal"
20 OUTPUT 717;":SENSe:PN1:CRANge ?"
30 ENTER 717;A$
```

SCPI.SENSe.PN(Ch).DCONverter.FREQuency

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PN(Ch).DCONverter.FREQuency = *Value*

Value = SCPI.SENSe.PN(Ch).DCONverter.FREQuency

Description

This command sets/gets downconverter input frequency (carrier nominal frequency), for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Downconverter input frequency (carrier nominal frequency)
Data Type	Double precision floating point type (Double)
Range	3G ~ 26.5G
Preset Value	3G
Unit	Hz
Resolution	100m
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

When the frequency band is 3G to 10GHz

Parameter	*<Double>*
Range	3G to 10G
***Preset* value**	3G
Unit	Hz
Resolution	100m

When the frequency band is 9G to 26.5GHz

Parameter	*<Double>*
Range	9G to 26.5G
Preset value	9G
Unit	Hz
Resolution	100m

Examples

```
Dim Var as Double
Var= 3000000000
SCPI.SENSe.PN1.DCONverter.FREQuency = Var
Var = SCPI.SENSe.PN1.DCONverter.FREQuency
```

Related Objects

SCPI.SENSe.PN(Ch).DCONverter.SSEarch.EXECute

Equivalent Key

PN Menu: **Setup** > **Nominal Frequency**

Equivalent SCPI Command

Syntax

:SENSe:PN[1-1]:DCONverter:FREQuency {3G ~ 26.5G}

:SENSe:PN[1-1]:DCONverter:FREQuency?

Query Response

{3G ~ 26.5G} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PN1:DCONverter:FREQuency 3000000000"
20 OUTPUT 717;":SENSe:PN1:DCONverter:FREQuency ?"
30 ENTER 717;A
```

SCPI.SENSe.PN(Ch).DCONverter.SSEarch.EXECute

Object Type

Method (**Write Only**)

Syntax

SCPI.SENSe.PN(Ch).DCONverter.SSEarch.EXECute

Description

This command searches carrier signal, for the selected channel *Ch*.

Examples

SCPI.SENSe.PN1.DCONverter.SSEarch.EXECute

Related Objects

SCPI.SENSe.PN(Ch).DCONverter.FREQuency

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SENSe:PN[1-1]:DCONverter:SSEarch:EXECute

Example of use

```
10 OUTPUT 717;":SENSe:PN1:DCONverter:SSEarch:EXECute"
```

SCPI.SENSe.PN(Ch).EPRescaler.DIVision

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PN(Ch).EPRescaler.DIVision = *Value*

Value = SCPI.SENSe.PN(Ch).EPRescaler.DIVision

Description

This command sets/gets external prescaler division, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	External prescaler division
Data Type	Long integer type (Long)
Range	1|2|4|8|16|32|64|128|256
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.SENSe.PN1.EPRescaler.DIVision = Var
Var = SCPI.SENSe.PN1.EPRescaler.DIVision
```

Related Objects

SCPI.SENSe.PN(Ch).EPRescaler.POWer

Equivalent Key

PN Menu: **System** > **Instrument Setup** > **PN Ext. Prescaler** > **Division**

SP Menu: **System** > **Instrument Setup** > **PN Ext. Prescaler** > **Division**

FP Menu: **System** > **Instrument Setup** > **PN Ext. Prescaler** > **Division**

TR Menu: **System** > **Instrument Setup** > **PN Ext. Prescaler** > **Division**

AM Menu: **System** > **Instrument Setup** > **PN Ext. Prescaler** > **Division**

BB Menu: **System** > **Instrument Setup** > **PN Ext. Prescaler** > **Division**

USER Menu: **System** > **Instrument Setup** > **PN Ext. Prescaler** > **Division**

Equivalent SCPI Command

Syntax

:SENSe:PN[1-1]:EPRescaler:DIVision {1|2|4|8|16|32|64|128|256}

:SENSe:PN[1-1]:EPRescaler:DIVision?

Query Response

{1|2|4|8|16|32|64|128|256} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PN1:EPRescaler:DIVision 1"
20 OUTPUT 717;":SENSe:PN1:EPRescaler:DIVision ?"
30 ENTER 717;A
```

SCPI.SENSe.PN(Ch).EPRescaler.POWer

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PN(Ch).EPRescaler.POWer = *Value*

Value = SCPI.SENSe.PN(Ch).EPRescaler.POWer

Description

This command sets/gets external prescaler output power (input power of RF1/RF2 of E5052), for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	External prescaler output power
Data Type	Double precision floating point type (Double)
Range	-15 ~ 0
Preset Value	0
Unit	dBm
Resolution	100u
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SENSe.PN1.EPRescaler.POWer = Var
Var = SCPI.SENSe.PN1.EPRescaler.POWer
```

Related Objects

SCPI.SENSe.PN(Ch).EPRescaler.DIVision

Equivalent Key

PN Menu: **System** > **Instrument Setup** > **PN Ext. Prescaler** > **Output Power Level**

SP Menu: **System** > **Instrument Setup** > **PN Ext. Prescaler** > **Output Power Level**

FP Menu: **System** > **Instrument Setup** > **PN Ext. Prescaler** > **Output Power Level**

TR Menu: **System** > **Instrument Setup** > **PN Ext. Prescaler** > **Output Power Level**

AM Menu: **System** > **Instrument Setup** > **PN Ext. Prescaler** > **Output Power Level**

BB Menu: **System** > **Instrument Setup** > **PN Ext. Prescaler** > **Output Power Level**

USER Menu: **System** > **Instrument Setup** > **PN Ext. Prescaler** > **Output Power Level**

Equivalent SCPI Command

Syntax

:SENSe:PN[1-1]:EPRescaler:POWer {-15 ~ 0}

:SENSe:PN[1-1]:EPRescaler:POWer?

Query Response

{-15 ~ 0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PN1:EPRescaler:POWer 0"
20 OUTPUT 717;":SENSe:PN1:EPRescaler:POWer ?"
30 ENTER 717;A
```

SCPI.SENSe.PN(Ch).FBANd

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PN(Ch).FBANd = *Value*

Value = SCPI.SENSe.PN(Ch).FBANd

Description

This command sets/gets frequency band, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Frequency band
Data Type	Character string type (String)
Range	BAND1|BAND2|BAND3|BAND4|BAND5|BAND6
Preset Value	BAND4
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

When the E5052B is used stand-alone, or with the downconverter turned off

Parameter	*<string>*
BAND1	Set frequency band to '10M - 41MHz'
BAND2	Set frequency band to '39M - 101MHz'
BAND3	Set frequency band to '99M - 1.5GHz'
BAND4(*Preset* value)	Set frequency band to '250M - 7GHz'

When the downconverter is turned on and with the RF input is set to ’E5052B Direct’

Parameter	*<string>*
BAND1	Set frequency band to '10M - 41MHz'
BAND2	Set frequency band to '39M - 101MHz'
BAND3	Set frequency band to '99M - 1.5GHz'
BAND4(Preset value)	Set frequency band to '250M - 3GHz'

When the downconverter is turned on and with the RF input is set to ’Downconverter’

Parameter	*<string>*
BAND5 (Preset Value)	Set frequency band to '3G - 10GHz'

BAND6	Set frequency band to '9G - 26.5GHz'

Examples

```
Dim Var as String
Var= "BAND1"
SCPI.SENSe.PN1.FBANd = Var
Var = SCPI.SENSe.PN1.FBANd
```

Equivalent Key

PN Menu: **Setup** > **Frequency Band**

Equivalent SCPI Command

Syntax

:SENSe:PN[1-1]:FBANd {BAND1|BAND2|BAND3|BAND4|BAND5|BAND6}

:SENSe:PN[1-1]:FBANd?

Query Response

{BAND1|BAND2|BAND3|BAND4|BAND5|BAND6} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PN1:FBANd BAND1"
20 OUTPUT 717;":SENSe:PN1:FBANd ?"
30 ENTER 717;A$
```

SCPI.SENSe.PN(Ch).FREQuency.STARt

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PN(Ch).FREQuency.STARt = *Value*

Value = SCPI.SENSe.PN(Ch).FREQuency.STARt

Description

This command sets/gets start frequency, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Start frequency
Data Type	Double precision floating point type (Double)
Range	1|10|100|1k
Preset Value	1k
Unit	Hz
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SENSe.PN1.FREQuency.STARt = Var
Var = SCPI.SENSe.PN1.FREQuency.STARt
```

Related Objects

SCPI.SENSe.PN(Ch).FREQuency.STOP

Equivalent Key

PN Menu: **Start** > **1Hz**

PN Menu: **Start** > **10Hz**

PN Menu: **Start** > **100Hz**

PN Menu: **Start** > **1kHz**

PN Menu: **Marker To** > **Marker** > **Start**

Equivalent SCPI Command

Syntax

:SENSe:PN[1-1]:FREQuency:STARt {1|10|100|1k}

:SENSe:PN[1-1]:FREQuency:STARt?

Query Response

{1|10|100|1k} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PN1:FREQuency:STARt 0"
20 OUTPUT 717;":SENSe:PN1:FREQuency:STARt ?"
30 ENTER 717;A
```

SCPI.SENSe.PN(Ch).FREQuency.STOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PN(Ch).FREQuency.STOP = *Value*

Value = SCPI.SENSe.PN(Ch).FREQuency.STOP

Description

This command sets/gets stop frequency, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Stop frequency
Data Type	Double precision floating point type (Double)
Range	100k|1M|5M|10M|20M|40M|100M
Preset Value	10M
Unit	Hz
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 100000
SCPI.SENSe.PN1.FREQuency.STOP = Var
Var = SCPI.SENSe.PN1.FREQuency.STOP
```

Related Objects

SCPI.SENSe.PN(Ch).FREQuency.STARt

Equivalent Key

PN Menu: **Stop** > **100kHz**

PN Menu: **Stop** > **1MHz**

PN Menu: **Stop** > **5MHz**

PN Menu: **Stop** > **10MHz**

PN Menu: **Stop** > **20MHz**

PN Menu: **Stop** > **40MHz**

PN Menu: **Stop** > **100MHz**

PN Menu: **Marker To** > **Marker** > **Stop**

Equivalent SCPI Command

Syntax

:SENSe:PN[1-1]:FREQuency:STOP {100k|1M|5M|10M|20M|40M|100M}

:SENSe:PN[1-1]:FREQuency:STOP?

Query Response

{100k|1M|5M|10M|20M|40M|100M} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PN1:FREQuency:STOP 100000"
20 OUTPUT 717;":SENSe:PN1:FREQuency:STOP ?"
30 ENTER 717;A
```

SCPI.SENSe.PN(Ch).IFGain

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PN(Ch).IFGain = *Value*

Value = SCPI.SENSe.PN(Ch).IFGain

Description

This command sets/gets IFGain 10dB Step, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	IFGain 10dB Step
Data Type	Double precision floating point type (Double)
Range	0 ~ 50
Preset Value	20
Unit	dB
Resolution	10
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SENSe.PN1.IFGain = Var
Var = SCPI.SENSe.PN1.IFGain
```

Equivalent Key

PN Menu: **Setup** > **IF Gain**

Equivalent SCPI Command

Syntax

:SENSe:PN[1-1]:IFGain {0 ~ 50}

:SENSe:PN[1-1]:IFGain?

Query Response

{0 ~ 50} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PN1:IFGain 0"
20 OUTPUT 717;":SENSe:PN1:IFGain ?"
30 ENTER 717;A
```

SCPI.SENSe.PN(Ch).LOBandwidth

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PN(Ch).LOBandwidth = *Value*

Value = SCPI.SENSe.PN(Ch).LOBandwidth

Description

This command sets/gets phase noise Local bandwidth optimization for the selected channel Ch.

Variable

Parameter	*Value*
Description	Phase noise Local bandwidth optimization
Data Type	Character string type (String)
Range	NARRow|WIDE
Preset Value	WIDE
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NARRow"
SCPI.SENSe.PN1.LOBandwidth = Var
Var = SCPI.SENSe.PN1.LOBandwidth
```

Equivalent Key

PN Menu: **Setup** > **LO PhNoise Optimize**

Equivalent SCPI Command

Syntax

:SENSe:PN[1-1]:LOBandwidth {NARRow|WIDE}

:SENSe:PN[1-1]:LOBandwidth?

Query Response

{NARRow|WIDE} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PN1:LOBandwidth NARRow"
20 OUTPUT 717;":SENSe:PN1:LOBandwidth ?"
30 ENTER 717;A$
```

SCPI.SENSe.PN(Ch).ROSCillator.BANDwidth

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PN(Ch).ROSCillator.BANDwidth = *Value*

Value = SCPI.SENSe.PN(Ch).ROSCillator.BANDwidth

Description

This command sets/gets bandwidth of reference oscillator, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Bandwidth of reference oscillator
Data Type	Character string type (String)
Range	NARRow|WIDE
Preset Value	NARRow
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NARRow"
SCPI.SENSe.PN1.ROSCillator.BANDwidth = Var
Var = SCPI.SENSe.PN1.ROSCillator.BANDwidth
```

Related Objects

SCPI.SENSe.PN(Ch).ROSCillator.REFerence(1-2).SOURce

Equivalent Key

PN Menu: **Setup** > **Reference Oscillator** > **Bandwidth**

Equivalent SCPI Command

Syntax

:SENSe:PN[1-1]:ROSCillator:BANDwidth {NARRow|WIDE}

:SENSe:PN[1-1]:ROSCillator:BANDwidth?

Query Response

{NARRow|WIDE} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PN1:ROSCillator:BANDwidth NARRow"
20 OUTPUT 717;":SENSe:PN1:ROSCillator:BANDwidth ?"
30 ENTER 717;A$
```

SCPI.SENSe.PN(Ch).ROSCillator.REFerence(1-2).SOURce

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PN(Ch).ROSCillator.REFerence(1-2).SOURce = *Value*

Value = SCPI.SENSe.PN(Ch).ROSCillator.REFerence(1-2).SOURce

Description

This command sets/gets source of reference oscillator, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Source of reference oscillator
Data Type	Character string type (String)
Range	INTernal|EXTernal
Preset Value	INTernal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "INTernal"
SCPI.SENSe.PN1.ROSCillator.REFerence2.SOURce = Var
Var = SCPI.SENSe.PN1.ROSCillator.REFerence2.SOURce
```

Related Objects

SCPI.SENSe.PN(Ch).ROSCillator.BANDwidth

Equivalent Key

PN Menu: **Setup** > **Reference Oscillator** > **Ref. Osc.1 Source**

Equivalent SCPI Command

Syntax

:SENSe:PN[1-1]:ROSCillator:REFerence[1-2]:SOURce {INTernal|EXTernal}

:SENSe:PN[1-1]:ROSCillator:REFerence[1-2]:SOURce?

Query Response

{INTernal|EXTernal} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PN1:ROSCillator:REFerence2:SOURce INTernal"
20 OUTPUT 717;":SENSe:PN1:ROSCillator:REFerence2:SOURce ?"
30 ENTER 717;A$
```

SCPI.SENSe.PN(Ch).SEGTable.MEASurement.QUALity

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PN(Ch).SEGTable.MEASurement.QUALity = *Value*

Value = SCPI.SENSe.PN(Ch).SEGTable.MEASurement.QUALity

Description

This command specifies the segment table to be used, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Specifies the segment table to be used
Data Type	Character string type (String)
Range	NORMal|FAST
Preset Value	NORMal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NORMal"
SCPI.SENSe.PN1.SEGTable.MEASurement.QUALity = Var
Var = SCPI.SENSe.PN1.SEGTable.MEASurement.QUALity
```

Equivalent Key

PN Menu: **Setup** > **Measurement Quality**

Equivalent SCPI Command

Syntax

:SENSe:PN[1-1]:SEGTable[:MEASurement][:QUALity] {NORMal|FAST}

:SENSe:PN[1-1]:SEGTable[:MEASurement][:QUALity]?

Query Response

{NORMal|FAST} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PN1:SEGTable[:MEASurement][:QUALity] NORMal"
20 OUTPUT 717;":SENSe:PN1:SEGTable[:MEASurement][:QUALity] ?"
30 ENTER 717;A$
```

SCPI.SENSe.PN(Ch).SWEep.POINts

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.SENSe.PN(Ch).SWEep.POINts

Description

This command gets number of points, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Number of points
Data Type	Long integer type (Long)
Range	2 ~ 1601
Preset Value	517
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 2
Var = SCPI.SENSe.PN1.SWEep.POINts
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SENSe:PN[1-1]:SWEep:POINts?

Query Response

{2 ~ 1601} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PN1:SWEep:POINts ?"
20 ENTER 717;A
```

SCPI.SENSe.PS(Ch).ASET

Object Type

Method (**Write Only**)

Syntax

SCPI.SENSe.PS(Ch).ASET

Description

This command sets Attn, Freq Band and IF Gain Auto Set, for the selected channel *Ch*.

Examples

SCPI.SENSe.PS1.ASET

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SENSe:PS[1-1]:ASET

Example of use

10 OUTPUT 717;":SENSe:PS1:ASET"

SCPI.SENSe.PS(Ch).AVERage.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.SENSe.PS(Ch).AVERage.CLEar

Description

This command clears average, for the selected channel *Ch*.

Examples

SCPI.SENSe.PS1.AVERage.CLEar

Related Objects

SCPI.SENSe.PS(Ch).AVERage.COUNt

SCPI.SENSe.PS(Ch).AVERage.STATe

Equivalent Key

PS Menu: **Average/BW** > **Averaging Restart**

Equivalent SCPI Command

Syntax

:SENSe:PS[1-1]:AVERage:CLEar

Example of use

```
10 OUTPUT 717;":SENSe:PS1:AVERage:CLEar"
```

SCPI.SENSe.PS(Ch).AVERage.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PS(Ch).AVERage.COUNt = *Value*

Value = SCPI.SENSe.PS(Ch).AVERage.COUNt

Description

This command sets/gets average count, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Average count
Data Type	Long integer type (Long)
Range	1 ~ 999
Preset Value	16
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.SENSe.PS1.AVERage.COUNt = Var
Var = SCPI.SENSe.PS1.AVERage.COUNt
```

Related Objects

SCPI.SENSe.PS(Ch).AVERage.CLEar

SCPI.SENSe.PS(Ch).AVERage.STATe

Equivalent Key

PS Menu: **Average/BW** > **Avg Factor**

Equivalent SCPI Command

Syntax

:SENSe:PS[1-1]:AVERage:COUNt {1 ~ 999}

:SENSe:PS[1-1]:AVERage:COUNt?

Query Response

{1 ~ 999} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PS1:AVERage:COUNt 1 "
20 OUTPUT 717;":SENSe:PS1:AVERage:COUNt?"
30 ENTER 717;A
```

SCPI.SENSe.PS(Ch).AVERage.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PS(Ch).AVERage.STATe = *Value*

Value = SCPI.SENSe.PS(Ch).AVERage.STATe

Description

This command sets/gets average ON/OFF, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Average (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.SENSe.PS1.AVERage.STATe = Var
Var = SCPI.SENSe.PS1.AVERage.STATe
```

Related Objects

SCPI.SENSe.PS(Ch).AVERage.CLEar

SCPI.SENSe.PS(Ch).AVERage.COUNt

Equivalent Key

PS Menu: **Average/BW** > **Averaging**

Equivalent SCPI Command

Syntax

:SENSe:PS[1-1]:AVERage:STATe {ON|OFF|1|0}

:SENSe:PS[1-1]:AVERage:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PS1:AVERage:STATe 1"
20 OUTPUT 717;":SENSe:PS1:AVERage:STATe?"
30 ENTER 717;A
```

SCPI.SENSe.PS(Ch).BANDwidth:RESolution

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PS(Ch).BANDwidth:RESolution = *Value*

Value = SCPI.SENSe.PS(Ch).BANDwidth:RESolution

Description

This command sets/gets RBW, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Resolution bandwidth
Data Type	Double precision floating point type (Double)
Range	95.4m ~ 25k
Preset Value	25k
Unit	Hz
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 1e3
SCPI.SENSe.PS1.BANDwidth:RESolution = Var
Var = SCPI.SENSe.PS1.BANDwidth:RESolution
```

Equivalent Key

PS Menu: **Average/BW** > **RBW**

Equivalent SCPI Command

Syntax

```
:SENSe:PS[1-1]:BANDwidth:RESolution {95.4m ~ 25k}
:SENSe:PS[1-1]:BANDwidth:RESolution?
```

Query Response

```
{95.4m ~ 25k} <newline><^END>
```

Example of use

```
10 OUTPUT 717;":SENSe:PS1:BANDwidth:RESolution 1e3"
20 OUTPUT 717;":SENSe:PS1:BANDwidth:RESolution?"
30 ENTER 717;A
```

SCPI.SENSe.PS(Ch).CORRelation.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PS(Ch).CORRelation.COUNt = *Value*

Value = SCPI.SENSe.PS(Ch).CORRelation.COUNt

Description

This command sets/gets correlation count, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Correlation count
Data Type	Double precision floating point type (Double)
Range	1 ~ 200k
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 1
SCPI.SENSe.PS1.CORRelation.COUNt = Var
Var = SCPI.SENSe.PS1.CORRelation.COUNt
```

Equivalent Key

PS Menu: **Average/BW** > **Correlation**

Equivalent SCPI Command

Syntax

```
:SENSe:PS[1-1]:CORRelation:COUNt {1 ~ 200k}
:SENSe:PS[1-1]:CORRelation:COUNt?
```

Query Response

```
{1 ~ 200k} <newline><^END>
```

Example of use

```
10 OUTPUT 717;":SENSe:PS1:CORRelation:COUNt 1 "
20 OUTPUT 717;":SENSe:PS1:CORRelation:COUNt?"
30 ENTER 717;A
```

SCPI.SENSe.PS(Ch).CRANge

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PS(Ch).CRANge = *Value*

Value = SCPI.SENSe.PS(Ch).CRANge

Description

This command sets/gets capture range, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Capture range
Data Type	Character string type (String)
Range	NORMal|WIDE
Preset Value	NORMal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NORMal"
SCPI.SENSe.PS1.CRANge = Var
Var = SCPI.SENSe.PS1.CRANge
```

Equivalent Key

PS Menu: **Setup** > **Capture Range**

Equivalent SCPI Command

Syntax

:SENSe:PS[1-1]:CRANge {NORMal|WIDE}

:SENSe:PS[1-1]:CRANge?

Query Response

{NORMal|WIDE} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PS1:CRANge NORMal"
20 OUTPUT 717;":SENSe:PS1:CRANge ?"
30 ENTER 717;A$
```

SCPI.SENSe.PS(Ch).DCONverter.FREQuency

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PS(Ch).DCONverter.FREQuency = *Value*

Value = SCPI.SENSe.PS(Ch).DCONverter.FREQuency

Description

This command sets/gets downconverter input frequency (carrier nominal frequency), for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Downconverter input frequency (carrier nominal frequency)
Data Type	Double precision floating point type (Double)
Range	3G ~ 26.5G
Preset Value	3G
Unit	Hz
Resolution	100m
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

When the frequency band is 3G to 10GHz

Parameter	*<Double>*
Range	3G to 10G
***Preset* value**	3G
Unit	Hz
Resolution	100m

When the frequency band is 9G to 26.5GHz

Parameter	*<Double>*
Range	9G to 26.5G
Preset value	9G
Unit	Hz
Resolution	100m

Examples

```
Dim Var as Double
Var= 3000000000
SCPI.SENSe.PS1.DCONverter.FREQuency = Var
Var = SCPI.SENSe.PS1.DCONverter.FREQuency
```

Related Objects

SCPI.SENSe.PS(Ch).DCONverter.SSEarch.EXECute

Equivalent Key

PS Menu: **Setup** > **Nominal Frequency**

Equivalent SCPI Command

Syntax

:SENSe:PS[1-1]:DCONverter:FREQuency {3G ~ 26.5G}

:SENSe:PS[1-1]:DCONverter:FREQuency?

Query Response

{3G ~ 26.5G} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PS1:DCONverter:FREQuency 3000000000"
20 OUTPUT 717;":SENSe:PS1:DCONverter:FREQuency ?"
30 ENTER 717;A
```

SCPI.SENSe.PS(Ch).DCONverter.SSEarch.EXECute

Object Type

Method (**Write Only**)

Syntax

SCPI.SENSe.PS(Ch).DCONverter.SSEarch.EXECute

Description

This command searches carrier signal, for the selected channel *Ch*.

Examples

SCPI.SENSe.PS1.DCONverter.SSEarch.EXECute

Related Objects

SCPI.SENSe.PS(Ch).DCONverter.FREQuency

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SENSe:PS[1-1]:DCONverter:SSEarch:EXECute

Example of use

```
10 OUTPUT 717;":SENSe:PS1:DCONverter:SSEarch:EXECute"
```

SCPI.SENSe.PS(Ch).FBANd

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PS(Ch).FBANd = *Value*

Value = SCPI.SENSe.PS(Ch).FBANd

Description

This command sets/gets frequency band, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Frequency band
Data Type	Character string type (String)
Range	BAND1|BAND2|BAND3|BAND4|BAND5|BAND6
Preset Value	BAND4
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

When the E5052B is used stand-alone, or with the downconverter turned off

Parameter	*<string>*
BAND1	Set frequency band to '10M - 41MHz'
BAND2	Set frequency band to '39M - 101MHz'
BAND3	Set frequency band to '99M - 1.5GHz'
BAND4(*Preset* value)	Set frequency band to '250M - 7GHz'

When the downconverter is turned on and with the *RF* input is set to 'E5052B Direct'

Parameter	*<string>*
BAND1	Set frequency band to '10M - 41MHz'
BAND2	Set frequency band to '39M - 101MHz'
BAND3	Set frequency band to '99M - 1.5GHz'
BAND4(Preset value)	Set frequency band to '250M - 3GHz'

When the downconverter is turned on and with the RF input is set to 'Downconverter'

Parameter	*<string>*
BAND5 (Preset Value)	Set frequency band to '3G - 10GHz'

BAND6	Set frequency band to '9G - 26.5GHz'

Examples

```
Dim Var as String
Var= "BAND1"
SCPI.SENSe.PS1.FBANd = Var
Var = SCPI.SENSe.PS1.FBANd
```

Equivalent Key

PS Menu: **Setup** > **Frequency Band**

Equivalent SCPI Command

Syntax

:SENSe:PS[1-1]:FBANd {BAND1|BAND2|BAND3|BAND4|BAND5|BAND6}

:SENSe:PS[1-1]:FBANd?

Query Response

{BAND1|BAND2|BAND3|BAND4|BAND5|BAND6} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PS1:FBANd BAND1"
20 OUTPUT 717;":SENSe:PS1:FBANd?"
30 ENTER 717;A$
```

SCPI.SENSe.PS(Ch).FREQuency.CENTer

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PS(Ch).FREQuency.CENTer = *Value*

Value = SCPI.SENSe.PS(Ch).FREQuency.CENTer

Description

This command sets/gets center frequency, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Center frequency
Data Type	Double precision floating point type (Double)
Range	1.1 ~ 99.9999998M
Preset Value	500.5k
Unit	Hz
Resolution	100m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 10e6
SCPI.SENSe.PS1.FREQuency.CENTer = Var
Var = SCPI.SENSe.PS1.FREQuency.CENTer
```

Related Objects

SCPI.SENSe.PS(Ch).FREQuency.SPAN

SCPI.SENSe.PS(Ch).FREQuency.STARt

SCPI.SENSe.PS(Ch).FREQuency.STOP

Equivalent Key

PS Menu: **Start/Center** > **Center**

PS Menu: **Stop/Span** > **Center**

PS Menu: **Marker To** > **Marker** -> **Center**

Equivalent SCPI Command

Syntax

:SENSe:PS[1-1]:FREQuency:CENTer {1.1 ~ 99.9999998M}

:SENSe:PS[1-1]:FREQuency:CENTer?

Query Response

{1.1 ~ 99.9999998M} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PS1:FREQuency:CENTer 10e6"
20 OUTPUT 717;":SENSe:PS1:FREQuency:CENTer ?"
30 ENTER 717;A
```

SCPI.SENSe.PS(Ch).FREQuency.SPAN

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PS(Ch).FREQuency.SPAN = *Value*

Value = SCPI.SENSe.PS(Ch).FREQuency.SPAN

Description

This command sets/gets span frequency, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Span frequency
Data Type	Double precision floating point type (Double)
Range	200m ~ 25M
Preset Value	999k
Unit	Hz
Resolution	200m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 10e6
SCPI.SENSe.PS1.FREQuency.SPAN = Var
Var = SCPI.SENSe.PS1.FREQuency.SPAN
```

Related Objects

SCPI.SENSe.PS(Ch).FREQuency.CENTer

SCPI.SENSe.PS(Ch).FREQuency.STARt

SCPI.SENSe.PS(Ch).FREQuency.STOP

Equivalent Key

PS Menu: **Start/Center** > **Span**

PS Menu: **Stop/Span** > **Span**

PS Menu: **Marker To** > **Marker** -> **Span**

Equivalent SCPI Command

Syntax

:SENSe:PS[1-1]:FREQuency:SPAN {200m ~ 25M }

:SENSe:PS[1-1]:FREQuency:SPAN?

Query Response

{200m ~ 25M } <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PS1:FREQuency:SPAN 10e6"
20 OUTPUT 717;":SENSe:PS1:FREQuency:SPAN ?"
30 ENTER 717;A
```

SCPI.SENSe.PS(Ch).FREQuency.STARt

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PS(Ch).FREQuency.STARt = *Value*

Value = SCPI.SENSe.PS(Ch).FREQuency.STARt

Description

This command sets/gets start frequency, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Start frequency
Data Type	Double precision floating point type (Double)
Range	1 ~ 99.9999998M
Preset Value	1k
Unit	Hz
Resolution	100m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 10e6
SCPI.SENSe.PS1.FREQuency.STARt = Var
Var = SCPI.SENSe.PS1.FREQuency.STARt
```

Related Objects

SCPI.SENSe.PS(Ch).FREQuency.STOP

SCPI.SENSe.PS(Ch).FREQuency.CENTer

SCPI.SENSe.PS(Ch).FREQuency.SPAN

Equivalent Key

PS Menu: **Start/Center** > **Start**

PS Menu: **Stop/Span** > **Start**

PS Menu: **Marker To** > **Marker** -> **Start**

Equivalent SCPI Command

Syntax

:SENSe:PS[1-1]:FREQuency:STARt {1 ~ 99.9999998M}

:SENSe:PS[1-1]:FREQuency:STARt?

Query Response

{1 ~ 99.9999998M} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PS1:FREQuency:STARt 10e6"
20 OUTPUT 717;":SENSe:PS1:FREQuency:STARt ?"
30 ENTER 717;A
```

SCPI.SENSe.PS(Ch).FREQuency.STOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PS(Ch).FREQuency.STOP = *Value*

Value = SCPI.SENSe.PS(Ch).FREQuency.STOP

Description

This command sets/gets stop frequency, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Stop frequency
Data Type	Double precision floating point type (Double)
Range	1.2 ~ 100M
Preset Value	1M
Unit	Hz
Resolution	100m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 100e3
SCPI.SENSe.PS1.FREQuency.STOP = Var
Var = SCPI.SENSe.PS1.FREQuency.STOP
```

Related Objects

SCPI.SENSe.PN(Ch).FREQuency.STARt

SCPI.SENSe.PS(Ch).FREQuency.CENTer

SCPI.SENSe.PS(Ch).FREQuency.SPAN

Equivalent Key

PS Menu: **Start/Center** > **Stop**

PS Menu: **Stop/Span** > **Stop**

PS Menu: **Marker To** > **Marker** -> **Stop**

Equivalent SCPI Command

Syntax

:SENSe:PS[1-1]:FREQuency:STOP {1.2 ~ 100M}

:SENSe:PS[1-1]:FREQuency:STOP?

Query Response

{1.2 ~ 100M} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PS1:FREQuency:STOP 100e3"
20 OUTPUT 717;":SENSe:PS1:FREQuency:STOP?"
30 ENTER 717;A
```

SCPI.SENSe.PS(Ch).IFGain

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PS(Ch).IFGain = *Value*

Value = SCPI.SENSe.PS(Ch).IFGain

Description

This command sets/gets IFGain 10dB Step, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	IFGain 10dB Step
Data Type	Double precision floating point type (Double)
Range	0 ~ 50
Preset Value	20
Unit	dB
Resolution	10
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SENSe.PS1.IFGain = Var
Var = SCPI.SENSe.PS1.IFGain
```

Equivalent Key

PS Menu: **Setup** > **IF Gain**

Equivalent SCPI Command

Syntax

```
:SENSe:PS[1-1]:IFGain {0 ~ 50}
:SENSe:PS[1-1]:IFGain?
```

Query Response

```
{0 ~ 50} <newline><^END>
```

Example of use

```
10 OUTPUT 717;":SENSe:PS1:IFGain 0"
20 OUTPUT 717;":SENSe:PS1:IFGain?"
30 ENTER 717;A
```

SCPI.SENSe.PS(Ch).LOBandwidth

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PS(Ch).LOBandwidth = *Value*

Value = SCPI.SENSe.PS(Ch).LOBandwidth

Description

This command sets/gets phase noise Local bandwidth optimization for the selected channel Ch.

Variable

Parameter	*Value*
Description	Phase noise Local bandwidth optimization
Data Type	Character string type (String)
Range	NARRow|WIDE
Preset Value	WIDE
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NARRow"
SCPI.SENSe.PS1.LOBandwidth = Var
Var = SCPI.SENSe.PS1.LOBandwidth
```

Equivalent Key

PS Menu: **Setup** > **LO PhNoise Optimize**

Equivalent SCPI Command

Syntax

:SENSe:PS[1-1]:LOBandwidth {NARRow|WIDE}

:SENSe:PS[1-1]:LOBandwidth?

Query Response

{NARRow|WIDE} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PS1:LOBandwidth NARRow"
20 OUTPUT 717;":SENSe:PS1:LOBandwidth?"
30 ENTER 717;A$
```

SCPI.SENSe.PS(Ch).ROSCillator.BANDwidth

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PS(Ch).ROSCillator.BANDwidth = *Value*

Value = SCPI.SENSe.PS(Ch).ROSCillator.BANDwidth

Description

This command sets/gets bandwidth of reference oscillator, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Bandwidth of reference oscillator
Data Type	Character string type (String)
Range	NARRow|WIDE
Preset Value	NARRow
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NARRow"
SCPI.SENSe.PS1.ROSCillator.BANDwidth = Var
Var = SCPI.SENSe.PS1.ROSCillator.BANDwidth
```

Related Objects

SCPI.SENSe.PS(Ch).ROSCillator.REFerence(1-2).SOURce

Equivalent Key

PS Menu: **Setup** > **Reference Oscillator** > **Bandwidth**

Equivalent SCPI Command

Syntax

:SENSe:PS[1-1]:ROSCillator:BANDwidth {NARRow|WIDE}

:SENSe:PS[1-1]:ROSCillator:BANDwidth?

Query Response

{NARRow|WIDE} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PS1:ROSCillator:BANDwidth NARRow"
20 OUTPUT 717;":SENSe:PS1:ROSCillator:BANDwidth?"
30 ENTER 717;A$
```

SCPI.SENSe.PS(Ch).ROSCillator.REFerence(1-2).SOURce

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.PS(Ch).ROSCillator.REFerence(1-2).SOURce = *Value*

Value = SCPI.SENSe.PS(Ch).ROSCillator.REFerence(1-2).SOURce

Description

This command sets/gets source of reference oscillator, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Source of reference oscillator
Data Type	Character string type (String)
Range	INTernal|EXTernal
Preset Value	INTernal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "INTernal"
SCPI.SENSe.PS1.ROSCillator.REFerence2.SOURce = Var
Var = SCPI.SENSe.PS1.ROSCillator.REFerence2.SOURce
```

Related Objects

SCPI.SENSe.PS(Ch).ROSCillator.BANDwidth

Equivalent Key

PS Menu: **Setup** > **Reference Oscillator** > **Ref. Osc.1 Source**

Equivalent SCPI Command

Syntax

:SENSe:PS[1-1]:ROSCillator:REFerence[1-2]:SOURce {INTernal|EXTernal}

:SENSe:PS[1-1]:ROSCillator:REFerence[1-2]:SOURce?

Query Response

{INTernal|EXTernal} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PS1:ROSCillator:REFerence2:SOURce INTernal"
20 OUTPUT 717;":SENSe:PS1:ROSCillator:REFerence2:SOURce?"
30 ENTER 717;A$
```

SCPI.SENSe.PS(Ch).SWEep.POINts

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.SENSe.PS(Ch).SWEep.POINts

Description

This command gets number of points, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Number of points
Data Type	Long integer type (Long)
Range	2 ~ 1601
Preset Value	17
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var = SCPI.SENSe.PS1.SWEep.POINts
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SENSe:PS[1-1]:SWEep:POINts?

Query Response

{2 ~ 1601} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:PS1:SWEep:POINts?"
20 ENTER 717;A
```

SCPI.SENSe.ROSCillator.REF1.SOURce

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.ROSCillator.REF1.SOURce = *Value*

Value = SCPI.SENSe.ROSCillator.REF1.SOURce

Description

This command sets/gets source of reference oscillator 1.

NOTE This command is available only in Phase Noise Measurement.

Variable

Parameter	*Value*
Description	Source of reference oscillator 1
Data Type	Character string type (String)
Range	INTernal|EXTernal
Preset Value	INTernal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "INTernal"
SCPI.SENSe.ROSCillator.REF1.SOURce = Var
Var = SCPI.SENSe.ROSCillator.REF1.SOURce
```

Related Objects

SCPI.SENSe.ROSCillator.SOURce

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SENSe:ROSCillator:REF1:SOURce {INTernal|EXTernal}

:SENSe:ROSCillator:REF1:SOURce?

Query Response

{INTernal|EXTernal} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:ROSCillator:REF1:SOURce INTernal"
20 OUTPUT 717;":SENSe:ROSCillator:REF1:SOURce ?"
30 ENTER 717;A$
```

SCPI.SENSe.ROSCillator.SOURce

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.SENSe.ROSCillator.SOURce

Description

This command gets source of reference oscillator.

Variable

Parameter	*Value*
Description	Source of reference oscillator
Data Type	Character string type (String)
Range	INTernal|EXTernal|UNLock
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as String
Var= "INTernal"
Var = SCPI.SENSe.ROSCillator.SOURce
```

Related Objects

SCPI.SENSe.ROSCillator.REF1.SOURce

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SENSe:ROSCillator:SOURce?

Query Response

{INTernal|EXTernal|UNLock} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:ROSCillator:SOURce ?"
20 ENTER 717;A$
```

SCPI.SENSe.SP(Ch).AVERage.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.SENSe.SP(Ch).AVERage.CLEar

Description

This command clears average, for the selected channel *Ch*.

Examples

SCPI.SENSe.SP1.AVERage.CLEar

Related Objects

SCPI.SENSe.SP(Ch).AVERage.CLEar

SCPI.SENSe.SP(Ch).AVERage.COUNt

SCPI.SENSe.SP(Ch).AVERage.STATe

SCPI.SENSe.SP(Ch).AVERage.TYPE

Equivalent Key

SP Menu: **Average/BW** > **Averaging Restart**

Equivalent SCPI Command

Syntax

:SENSe:SP[1-1]:AVERage:CLEar

Example of use

```
10 OUTPUT 717;":SENSe:SP1:AVERage:CLEar"
```

SCPI.SENSe.SP(Ch).AVERage.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.SP(Ch).AVERage.COUNt = *Value*

Value = SCPI.SENSe.SP(Ch).AVERage.COUNt

Description

This command sets/gets average count, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Average count
Data Type	Long integer type (Long)
Range	1 ~ 999
Preset Value	16
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.SENSe.SP1.AVERage.COUNt = Var
Var = SCPI.SENSe.SP1.AVERage.COUNt
```

Related Objects

SCPI.SENSe.SP(Ch).AVERage.CLEar

SCPI.SENSe.SP(Ch).AVERage.STATe

SCPI.SENSe.SP(Ch).AVERage.TYPE

Equivalent Key

SP Menu: **Average/BW** > **Avg Factor**

Equivalent SCPI Command

Syntax

:SENSe:SP[1-1]:AVERage:COUNt {1 ~ 999}

:SENSe:SP[1-1]:AVERage:COUNt?

Query Response

{1 ~ 999} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:SP1:AVERage:COUNt 1 "
20 OUTPUT 717;":SENSe:SP1:AVERage:COUNt ?"
30 ENTER 717;A
```

SCPI.SENSe.SP(Ch).AVERage.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.SP(Ch).AVERage.STATe = *Value*

Value = SCPI.SENSe.SP(Ch).AVERage.STATe

Description

This command sets/gets average ON/OFF, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Average (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.SENSe.SP1.AVERage.STATe = Var
Var = SCPI.SENSe.SP1.AVERage.STATe
```

Related Objects

SCPI.SENSe.SP(Ch).AVERage.CLEar

SCPI.SENSe.SP(Ch).AVERage.COUNt

SCPI.SENSe.SP(Ch).AVERage.TYPE

Equivalent Key

SP Menu: **Average/BW** > **Averaging**

Equivalent SCPI Command

Syntax

:SENSe:SP[1-1]:AVERage:STATe {ON|OFF|1|0}

:SENSe:SP[1-1]:AVERage:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:SP1:AVERage:STATe 1"
20 OUTPUT 717;":SENSe:SP1:AVERage:STATe ?"
30 ENTER 717;A
```

SCPI.SENSe.SP(Ch).AVERage.TYPE

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.SP(Ch).AVERage.TYPE = *Value*

Value = SCPI.SENSe.SP(Ch).AVERage.TYPE

Description

This command sets/gets average mode, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Average mode
Data Type	Character string type (String)
Range	RMS|LOGarithmic
Preset Value	LOGarithmic
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "RMS"
SCPI.SENSe.SP1.AVERage.TYPE = Var
Var = SCPI.SENSe.SP1.AVERage.TYPE
```

Related Objects

SCPI.SENSe.SP(Ch).AVERage.CLEar

SCPI.SENSe.SP(Ch).AVERage.COUNt

SCPI.SENSe.SP(Ch).AVERage.STATe

Equivalent Key

SP Menu: **Average/BW** > **Averaging Type**

Equivalent SCPI Command

Syntax

:SENSe:SP[1-1]:AVERage:TYPE {RMS|LOGarithmic}

:SENSe:SP[1-1]:AVERage:TYPE?

Query Response

{RMS|LOGarithmic} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:SP1:AVERage:TYPE RMS"
20 OUTPUT 717;":SENSe:SP1:AVERage:TYPE ?"
30 ENTER 717;A$
```

SCPI.SENSe.SP(Ch).BANDwidth.RESolution

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.SP(Ch).BANDwidth.RESolution = *Value*

Value = SCPI.SENSe.SP(Ch).BANDwidth.RESolution

Description

This command sets/gets bandwidth resolution, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Bandwidth resolution
Data Type	Double precision floating point type (Double)
Range	1.53 ~ 1.6M
Preset Value	25k
Unit	Hz
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SENSe.SP1.BANDwidth.RESolution = Var
Var = SCPI.SENSe.SP1.BANDwidth.RESolution
```

Equivalent Key

SP Menu: **Average/BW** > **RBW**

Equivalent SCPI Command

Syntax

:SENSe:SP[1-1]:BANDwidth:RESolution {1.53 ~ 1.6M}

:SENSe:SP[1-1]:BANDwidth:RESolution?

Query Response

{1.53 ~ 1.6M} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:SP1:BANDwidth:RESolution 0"
20 OUTPUT 717;":SENSe:SP1:BANDwidth:RESolution ?"
30 ENTER 717;A
```

SCPI.SENSe.SP(Ch).CARRier.FBANd

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.SP(Ch).CARRier.FBANd = *Value*

Value = SCPI.SENSe.SP(Ch).CARRier.FBANd

Description

This command sets/gets carrier frequency band, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Carrier frequency band
Data Type	Character string type (String)
Range	LOW|HIGH|BAND3|BAND4
Preset Value	HIGH
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

When the E5052B is used stand-alone, or with the downconverter turned off, or with the downconverter on and the *RF* input is set to 'E5052B Direct'

Parameter	*<string>*
LOW	Set the carrier frequency band to '10M-1.5GHz'
HIGH (*Preset* value)	Set the carrier frequency band to '300M-7GHz'

When the downconverter is turned on and the RF input is set to 'Downconverter'

Parameter	*<string>*
BAND3 (Preset value)	Set the carrier frequency band to '3G-10GHz'
BAND4	Set the carrier frequency band to '9G-26.5GHz'

Examples

```
Dim Var as String
Var= "LOW"
SCPI.SENSe.SP1.CARRier.FBANd = Var
Var = SCPI.SENSe.SP1.CARRier.FBANd
```

Related Objects

SCPI.SENSe.SP(Ch).CARRier.SET.CENTer

Equivalent Key

SP Menu: **Start/Center** > **Carrier To** > **Frequency Band**

SP Menu: **Stop/Span** > **Carrier To** > **Frequency Band**

Equivalent SCPI Command

Syntax

:SENSe:SP[1-1]:CARRier:FBANd {LOW|HIGH|BAND3|BAND4}

:SENSe:SP[1-1]:CARRier:FBANd?

Query Response

{LOW|HIGH|BAND3|BAND4} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:SP1:CARRier:FBANd LOW"
20 OUTPUT 717;":SENSe:SP1:CARRier:FBANd ?"
30 ENTER 717;A$
```

SCPI.SENSe.SP(Ch).CARRier.SET.CENTer

Object Type

Method (**Write Only**)

Syntax

SCPI.SENSe.SP(Ch).CARRier.SET.CENTer = *Value*

Description

This command sets carrier freq x number to center freq, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Carrier freq x number to center freq
Data Type	Double precision floating point type (Double)
Range	1 ~ 20
Preset Value	-
Unit	-
Resolution	1
Preset Effect	No
Save-Recall Effect	No
Error	110, "Target freq out of range" 111, "No signal found"

Examples

```
Dim Var as Double
Var= 1
SCPI.SENSe.SP1.CARRier.SET.CENTer = Var
```

Related Objects

SCPI.SENSe.SP(Ch).CARRier.FBANd

Equivalent Key

SP Menu : **Start/Center** > **Carrier To** > **Carrier -> Center**

SP Menu : **Start/Center** > **Carrier To** > **Carrier x2 -> Center**

SP Menu : **Start/Center** > **Carrier To** > **Carrier x3 -> Center**

Equivalent SCPI Command

Syntax

:SENSe:SP[1-1]:CARRier:SET:CENTer {1 ~ 20}

Example of use

```
10 OUTPUT 717;":SENSe:SP1:CARRier:SET:CENTer 1"
20 ENTER 717;A
```

SCPI.SENSe.SP(Ch).DETector.FUNCtion

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.SP(Ch).DETector.FUNCtion = *Value*

Value = SCPI.SENSe.SP(Ch).DETector.FUNCtion

Description

This command sets/gets detect mode, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Detect Mode
Data Type	Character string type (String)
Range	POSitive|SAMPle
Preset Value	POSitive
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "POSitive"
SCPI.SENSe.SP1.DETector.FUNCtion = Var
Var = SCPI.SENSe.SP1.DETector.FUNCtion
```

Equivalent Key

SP Menu: **Format** > **Detector Mode**

Equivalent SCPI Command

Syntax

:SENSe:SP[1-1]:DETector:FUNCtion {POSitive|SAMPle}

:SENSe:SP[1-1]:DETector:FUNCtion?

Query Response

{POSitive|SAMPle} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:SP1:DETector:FUNCtion POSitive"
20 OUTPUT 717;":SENSe:SP1:DETector:FUNCtion ?"
30 ENTER 717;A$
```

SCPI.SENSe.SP(Ch).FREQuency.CENTer

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.SP(Ch).FREQuency.CENTer = *Value*

Value = SCPI.SENSe.SP(Ch).FREQuency.CENTer

Description

This command sets/gets center frequency, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Center frequency
Data Type	Double precision floating point type (Double)
Range	10M ~ 337G
Preset Value	1G
Unit	Hz
Resolution	100m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 10000000
SCPI.SENSe.SP1.FREQuency.CENTer = Var
Var = SCPI.SENSe.SP1.FREQuency.CENTer
```

Related Objects

SCPI.SENSe.SP(Ch).FREQuency.CENTer

SCPI.SENSe.SP(Ch).FREQuency.SPAN

SCPI.SENSe.SP(Ch).FREQuency.STARt

SCPI.SENSe.SP(Ch).FREQuency.STOP

Equivalent Key

SP Menu: **Start/Center** > **Center**

SP Menu: **Stop/Span** > **Center**

SP Menu: **Marker To** > **Marker** > **Center**

Equivalent SCPI Command

Syntax

:SENSe:SP[1-1]:FREQuency:CENTer {10M ~ 337G}

:SENSe:SP[1-1]:FREQuency:CENTer?

Query Response

{10M ~ 337G} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:SP1:FREQuency:CENTer 10000000"
20 OUTPUT 717;":SENSe:SP1:FREQuency:CENTer ?"
30 ENTER 717;A
```

SCPI.SENSe.SP(Ch).FREQuency.SPAN

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.SP(Ch).FREQuency.SPAN = *Value*

Value = SCPI.SENSe.SP(Ch).FREQuency.SPAN

Description

This command sets/gets frequency span, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Frequency span
Data Type	Double precision floating point type (Double)
Range	100 ~ 15M
Preset Value	15M
Unit	Hz
Resolution	200m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SENSe.SP1.FREQuency.SPAN = Var
Var = SCPI.SENSe.SP1.FREQuency.SPAN
```

Related Objects

SCPI.SENSe.SP(Ch).FREQuency.CENTer

SCPI.SENSe.SP(Ch).FREQuency.STARt

SCPI.SENSe.SP(Ch).FREQuency.STOP

Equivalent Key

SP Menu: **Start/Center** > **Span**

SP Menu: **Stop/Span** > **Span**

Equivalent SCPI Command

Syntax

:SENSe:SP[1-1]:FREQuency:SPAN {100 ~ 15M}

:SENSe:SP[1-1]:FREQuency:SPAN?

Query Response

{100 ~ 15M} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:SP1:FREQuency:SPAN 0"
20 OUTPUT 717;":SENSe:SP1:FREQuency:SPAN ?"
30 ENTER 717;A
```

SCPI.SENSe.SP(Ch).FREQuency.STARt

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.SP(Ch).FREQuency.STARt = *Value*

Value = SCPI.SENSe.SP(Ch).FREQuency.STARt

Description

This command sets/gets start frequency, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Start frequency
Data Type	Double precision floating point type (Double)
Range	9M ~ 336.99999995G
Preset Value	992.5M
Unit	Hz
Resolution	100m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 9000000
SCPI.SENSe.SP1.FREQuency.STARt = Var
Var = SCPI.SENSe.SP1.FREQuency.STARt
```

Related Objects

SCPI.SENSe.SP(Ch).FREQuency.CENTer

SCPI.SENSe.SP(Ch).FREQuency.SPAN

SCPI.SENSe.SP(Ch).FREQuency.STOP

Equivalent Key

SP Menu: **Start/Center** > **Start**

SP Menu: **Stop/Span** > **Start**

SP Menu: **Marker To** > **Marker** > **Start**

Equivalent SCPI Command

Syntax

:SENSe:SP[1-1]:FREQuency:STARt {9M ~ 336.99999995G}

:SENSe:SP[1-1]:FREQuency:STARt?

Query Response

{9M ~ 336.99999995G} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:SP1:FREQuency:STARt 9000000"
20 OUTPUT 717;":SENSe:SP1:FREQuency:STARt ?"
30 ENTER 717;A
```

SCPI.SENSe.SP(Ch).FREQuency.STOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.SP(Ch).FREQuency.STOP = *Value*

Value = SCPI.SENSe.SP(Ch).FREQuency.STOP

Description

This command sets/gets stop frequency, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Stop frequency
Data Type	Double precision floating point type (Double)
Range	10.00005M ~ 337.0075G
Preset Value	1.0075G
Unit	Hz
Resolution	100m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 10000050
SCPI.SENSe.SP1.FREQuency.STOP = Var
Var = SCPI.SENSe.SP1.FREQuency.STOP
```

Related Objects

SCPI.SENSe.SP(Ch).FREQuency.CENTer

SCPI.SENSe.SP(Ch).FREQuency.SPAN

SCPI.SENSe.SP(Ch).FREQuency.STARt

Equivalent Key

SP Menu: **Start/Center** > **Stop**

SP Menu: **Stop/Span** > **Stop**

SP Menu: **Marker To** > **Marker** > **Stop**

Equivalent SCPI Command

Syntax

:SENSe:SP[1-1]:FREQuency:STOP {10.00005M ~ 337.0075G}

:SENSe:SP[1-1]:FREQuency:STOP?

Query Response

{10.00005M ~ 337.0075G} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:SP1:FREQuency:STOP 10000050"
20 OUTPUT 717;":SENSe:SP1:FREQuency:STOP ?"
30 ENTER 717;A
```

SCPI.SENSe.SP(Ch).POWer.RLEVel

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.SP(Ch).POWer.RLEVel = *Value*

Value = SCPI.SENSe.SP(Ch).POWer.RLEVel

Description

This command sets/gets reference level, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Reference level
Data Type	Double precision floating point type (Double)
Range	-45 ~ 30
Preset Value	5
Unit	dBm
Resolution	5
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SENSe.SP1.POWer.RLEVel = Var
Var = SCPI.SENSe.SP1.POWer.RLEVel
```

Equivalent Key

SP Menu: **Setup** > **Reference Level**

Equivalent SCPI Command

Syntax

:SENSe:SP[1-1]:POWer:RLEVel {-45 ~ 30}

:SENSe:SP[1-1]:POWer:RLEVel?

Query Response

{-45 ~ 30} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:SP1:POWer:RLEVel 0"
20 OUTPUT 717;":SENSe:SP1:POWer:RLEVel ?"
30 ENTER 717;A
```

SCPI.SENSe.SP(Ch).SWEep.POINts

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.SENSe.SP(Ch).SWEep.POINts

Description

This command gets number of points, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Number of points
Data Type	Long integer type (Long)
Range	2 ~ 1024
Preset Value	154
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 2
Var = SCPI.SENSe.SP1.SWEep.POINts
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SENSe:SP[1-1]:SWEep:POINts?

Query Response

{2 ~ 1024} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:SP1:SWEep:POINts ?"
20 ENTER 717;A
```

SCPI.SENSe.TR(Ch).AVERage.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.SENSe.TR(Ch).AVERage.CLEar

Description

This command clears average, for the selected channel *Ch*.

Examples

SCPI.SENSe.TR1.AVERage.CLEar

Related Objects

SCPI.SENSe.TR(Ch).AVERage.CLEar

SCPI.SENSe.TR(Ch).AVERage.COUNt

SCPI.SENSe.TR(Ch).AVERage.STATe

Equivalent Key

TR Menu: **Average** > **Averaging Restart**

Equivalent SCPI Command

Syntax

:SENSe:TR[1-1]:AVERage:CLEar

Example of use

```
10 OUTPUT 717;":SENSe:TR1:AVERage:CLEar"
```

SCPI.SENSe.TR(Ch).AVERage.COUNt

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.TR(Ch).AVERage.COUNt = *Value*

Value = SCPI.SENSe.TR(Ch).AVERage.COUNt

Description

This command sets/gets average count, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Average count
Data Type	Long integer type (Long)
Range	1 ~ 999
Preset Value	16
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.SENSe.TR1.AVERage.COUNt = Var
Var = SCPI.SENSe.TR1.AVERage.COUNt
```

Related Objects

SCPI.SENSe.TR(Ch).AVERage.CLEar

SCPI.SENSe.TR(Ch).AVERage.STATe

Equivalent Key

TR Menu: **Average** > **Avg Factor**

Equivalent SCPI Command

Syntax

:SENSe:TR[1-1]:AVERage:COUNt {1 ~ 999}

:SENSe:TR[1-1]:AVERage:COUNt?

Query Response

{1 ~ 999} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:TR1:AVERage:COUNt 1 "
20 OUTPUT 717;":SENSe:TR1:AVERage:COUNt ?"
30 ENTER 717;A
```

SCPI.SENSe.TR(Ch).AVERage.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.TR(Ch).AVERage.STATe = *Value*

Value = SCPI.SENSe.TR(Ch).AVERage.STATe

Description

This command sets/gets average ON/OFF, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Average (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.SENSe.TR1.AVERage.STATe = Var
Var = SCPI.SENSe.TR1.AVERage.STATe
```

Related Objects

SCPI.SENSe.TR(Ch).AVERage.CLEar

SCPI.SENSe.TR(Ch).AVERage.COUNt

Equivalent Key

TR Menu: **Average** > **Averaging**

Equivalent SCPI Command

Syntax

:SENSe:TR[1-1]:AVERage:STATe {ON|OFF|1|0}

:SENSe:TR[1-1]:AVERage:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:TR1:AVERage:STATe 1"
20 OUTPUT 717;":SENSe:TR1:AVERage:STATe ?"
30 ENTER 717;A
```

SCPI.SENSe.TR(Ch).MMODe

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.TR(Ch).MMODe = *Value*

Value = SCPI.SENSe.TR(Ch).MMODe

Description

This command sets/gets measurement mode, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Measurement mode
Data Type	Character string type (String)
Range	WN|NN
Preset Value	WN
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NN"
SCPI.SENSe.TR1.MMODe = Var
Var = SCPI.SENSe.TR1.MMODe
```

Equivalent Key

TR Menu: **Setup** > **Measurement Mode**

Equivalent SCPI Command

Syntax

:SENSe:TR[1-1]:MMODe {WN|NN}

:SENSe:TR[1-1]:MMODe?

Query Response

{WN|NN} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:TR1:MMODe NN"
20 OUTPUT 717;":SENSe:TR1:MMODe?"
30 ENTER 717;A$
```

SCPI.SENSe.TR(Ch).NARRow(Nr).AMODe

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.TR(Ch).NARRow(Nr).AMODe = *Value*

Value = SCPI.SENSe.TR(Ch).NARRow(Nr).AMODe

Description

This command sets/gets narrow advanced setup mode, for the selected narrow *Nr* of the selected channel *Ch*.

NOTE This command is available when narrow 1.

Variable

Parameter	*Value*
Description	Narrow advanced setup mode
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.SENSe.TR1.NARRow1.AMODe = Var
Var = SCPI.SENSe.TR1.NARRow1.AMODe
```

Equivalent Key

TR Menu: **Setup** > **Advanced Setup** > **Advanced mode**

Equivalent SCPI Command

Syntax

:SENSe:TR[1-1]:NARRow[1-2]:AMODe {ON|OFF|1|0}

:SENSe:TR[1-1]:NARRow[1-2]:AMODe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:TR1:NARRow1:AMODe 1"
20 OUTPUT 717;":SENSe:TR1:NARRow1:AMODe?"
30 ENTER 717;A
```

SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.PREFerence

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.PREFerence = *Value*

Value = SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.PREFerence

Description

This command sets/gets phase reference frequency, for the selected narrow *Nr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Phase reference frequency
Data Type	Double precision floating point type (Double)
Range	9.2M ~ 337.04G
Preset Value	1G
Unit	Hz
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 9200000
SCPI.SENSe.TR1.NARRow1.FREQuency.PREFerence = Var
Var = SCPI.SENSe.TR1.NARRow1.FREQuency.PREFerence
```

Related Objects

SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.RANGe

SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.RESolution

SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.TARGet

Equivalent Key

TR Menu: **Setup** > **Phase Reference**

TR Menu: **Setup** > **N2 Phase Reference**

TR Menu: **Marker To** > **Marker** > **Phase Reference**

TR Menu: **Marker To** > **Marker** > **N2 Phase Reference**

Equivalent SCPI Command

Syntax

:SENSe:TR[1-1]:NARRow[1-2]:FREQuency:PREFerence {9.2M ~ 337.04G}

:SENSe:TR[1-1]:NARRow[1-2]:FREQuency:PREFerence?

Query Response

{9.2M ~ 337.04G} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:TR1:NARRow1:FREQuency:PREFerence 9200000"
20 OUTPUT 717;":SENSe:TR1:NARRow1:FREQuency:PREFerence?"
30 ENTER 717;A
```

SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.RANGe

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.RANGe = *Value*

Value = SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.RANGe

Description

This command sets/gets frequency span, for the selected narrow *Nr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Frequency span
Data Type	Character string type (String)
Range	R80|R25_6|R1_6|R0_2|R25K|R3K
Preset Value	R25_6
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "R80"
SCPI.SENSe.TR1.NARRow1.FREQuency.RANGe = Var
Var = SCPI.SENSe.TR1.NARRow1.FREQuency.RANGe
```

Related Objects

SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.PREFerence

SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.RESolution

SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.TARGet

Equivalent Key

TR Menu: **Setup** > **Freq Range**

TR Menu: **Setup** > **N2 Freq Range**

Equivalent SCPI Command

Syntax

:SENSe:TR[1-1]:NARRow[1-2]:FREQuency:RANGe {R80|R25_6|R1_6|R0_2|R25K|R3K}

:SENSe:TR[1-1]:NARRow[1-2]:FREQuency:RANGe?

Query Response

{R80|R25_6|R1_6|R0_2|R25K|R3K} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:TR1:NARRow1:FREQuency:RANGe R80"
20 OUTPUT 717;":SENSe:TR1:NARRow1:FREQuency:RANGe?"
30 ENTER 717;A$
```

SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.RESolution

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.RESolution = *Value*

Value = SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.RESolution

Description

This command sets/gets Frequency resolution, for the selected narrow *Nr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Frequency resolution
Data Type	Character string type (String)
Range	R1|R2|R3|R4
Preset Value	R1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "R1"
SCPI.SENSe.TR1.NARRow1.FREQuency.RESolution = Var
Var = SCPI.SENSe.TR1.NARRow1.FREQuency.RESolution
```

Related Objects

SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.PREFerence

SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.RANGe

SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.TARGet

Equivalent Key

TR Menu: **Setup** > **Advanced Setup** > **Narrow Freq Reso.**

Equivalent SCPI Command

Syntax

:SENSe:TR[1-1]:NARRow[1-2]:FREQuency:RESolution {R1|R2|R3|R4}

:SENSe:TR[1-1]:NARRow[1-2]:FREQuency:RESolution?

Query Response

{R1|R2|R3|R4} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:TR1:NARRow1:FREQuency:RESolution R1"
20 OUTPUT 717;":SENSe:TR1:NARRow1:FREQuency:RESolution?"
30 ENTER 717;A$
```

SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.TARGet

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.TARGet = *Value*

Value = SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.TARGet

Description

This command sets/gets target frequency, for the selected narrow *Nr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Target frequency
Data Type	Double precision floating point type (Double)
Range	10M ~ 337G
Preset Value	1G
Unit	Hz
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 10000000
SCPI.SENSe.TR1.NARRow1.FREQuency.TARGet = Var
Var = SCPI.SENSe.TR1.NARRow1.FREQuency.TARGet
```

Related Objects

SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.PREFerence

SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.RANGe

SCPI.SENSe.TR(Ch).NARRow(Nr).FREQuency.RESolution

Equivalent Key

TR Menu: **Setup** > **Target Freq**

TR Menu: **Setup** > **N2 Target Freq**

TR Menu: **Marker To** > **Marker** > **Target Freq**

TR Menu: **Marker To** > **Marker** > **N2 Target Freq**

Equivalent SCPI Command

Syntax

:SENSe:TR[1-1]:NARRow[1-2]:FREQuency:TARGet {10M ~ 337G}

:SENSe:TR[1-1]:NARRow[1-2]:FREQuency:TARGet?

Query Response

{10M ~ 337G} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:TR1:NARRow1:FREQuency:TARGet 10000000"
20 OUTPUT 717;":SENSe:TR1:NARRow1:FREQuency:TARGet?"
30 ENTER 717;A
```

SCPI.SENSe.TR(Ch).NARRow(Nr).SWEep.POINts

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.SENSe.TR(Ch).NARRow(Nr).SWEep.POINts

Description

This command gets number of points, for the selected narrow *Nr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Number of points
Data Type	Long integer type (Long)
Range	626 ~ 1564
Preset Value	801
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 626
Var = SCPI.SENSe.TR1.NARRow1.SWEep.POINts
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SENSe:TR[1-1]:NARRow[1-2]:SWEep:POINts?

Query Response

{626 ~ 1564} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:TR1:NARRow1:SWEep:POINts?"
20 ENTER 717;A
```

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.DSTart

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.DSTart = *Value*

Value = SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.DSTart

Description

This command sets/gets narrow display start value, for the selected narrow *Nr* of the selected channel *Ch*.

NOTE This command is available when narrow 1.

Variable

Parameter	*Value*
Description	Narrow display start value
Data Type	Double precision floating point type (Double)
Range	-20 ~ 1.00107935999k
Preset Value	-50m
Unit	s
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SENSe.TR1.NARRow1.TIME.DSTart = Var
Var = SCPI.SENSe.TR1.NARRow1.TIME.DSTart
```

Related Objects

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.OFFSet

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.REFerence

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.RESolution

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.SPAN

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.ZOUT

Equivalent Key

TR Menu: **Setup** > **Advanced Setup** > **Narrow Display Start**

Equivalent SCPI Command

Syntax

:SENSe:TR[1-1]:NARRow[1-2]:TIME:DSTart {-20 ~ 1.00107935999k}

:SENSe:TR[1-1]:NARRow[1-2]:TIME:DSTart?

Query Response

{-20 ~ 1.00107935999k} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:TR1:NARRow1:TIME:DSTart 0"
20 OUTPUT 717;":SENSe:TR1:NARRow1:TIME:DSTart?"
30 ENTER 717;A
```

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.OFFSet

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.OFFSet = *Value*

Value = SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.OFFSet

Description

This command sets/gets offset for reference point, for the selected narrow *Nr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Offset for reference point
Data Type	Double precision floating point type (Double)
Range	-20 ~ 1.00107936k
Preset Value	0
Unit	s
Resolution	1n
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SENSe.TR1.NARRow1.TIME.OFFSet = Var
Var = SCPI.SENSe.TR1.NARRow1.TIME.OFFSet
```

Related Objects

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.DSTart

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.REFerence

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.RESolution

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.SPAN

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.ZOUT

Equivalent Key

TR Menu: **Span** > **Narrow Time Offset**

TR Menu: **Span** > **Narrow2 Time Offset**

Equivalent SCPI Command

Syntax

:SENSe:TR[1-1]:NARRow[1-2]:TIME:OFFSet {-20 ~ 1.00107936k}

:SENSe:TR[1-1]:NARRow[1-2]:TIME:OFFSet?

Query Response

{-20 ~ 1.00107936k} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:TR1:NARRow1:TIME:OFFSet 0"
20 OUTPUT 717;":SENSe:TR1:NARRow1:TIME:OFFSet?"
30 ENTER 717;A
```

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.REFerence

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.REFerence = *Value*

Value = SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.REFerence

Description

This command sets/gets reference position for span, for the selected narrow *Nr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Reference position for span
Data Type	Character string type (String)
Range	LEFT|CENTer|RIGHt
Preset Value	CENTer
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "LEFT"
SCPI.SENSe.TR1.NARRow1.TIME.REFerence = Var
Var = SCPI.SENSe.TR1.NARRow1.TIME.REFerence
```

Related Objects

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.DSTart

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.OFFSet

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.RESolution

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.SPAN

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.ZOUT

Equivalent Key

TR Menu: **Span** > **Narrow Ref Position**

TR Menu: **Span** > **Narrow2 Ref Position**

Equivalent SCPI Command

Syntax

:SENSe:TR[1-1]:NARRow[1-2]:TIME:REFerence {LEFT|CENTer|RIGHt}

:SENSe:TR[1-1]:NARRow[1-2]:TIME:REFerence?

Query Response

{LEFT|CENTer|RIGHt} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:TR1:NARRow1:TIME:REFerence LEFT"
20 OUTPUT 717;":SENSe:TR1:NARRow1:TIME:REFerence?"
30 ENTER 717;A$
```

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.RESolution

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.RESolution = *Value*

Value = SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.RESolution

Description

This command sets/gets narrow time resolution, for the selected narrow *Nr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Narrow time resolution
Data Type	Double precision floating point type (Double)
Range	8n ~ 100.007936m
Preset Value	100u
Unit	s
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SENSe.TR1.NARRow1.TIME.RESolution = Var
Var = SCPI.SENSe.TR1.NARRow1.TIME.RESolution
```

Related Objects

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.DSTart

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.OFFSet

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.REFerence

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.SPAN

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.ZOUT

Equivalent Key

TR Menu: **Setup** > **Advanced Setup** > **Narrow Time Reso.**

Equivalent SCPI Command

Syntax

:SENSe:TR[1-1]:NARRow[1-2]:TIME:RESolution {8n ~ 100.007936m}

:SENSe:TR[1-1]:NARRow[1-2]:TIME:RESolution?

Query Response

{8n ~ 100.007936m} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:TR1:NARRow1:TIME:RESolution 0"
20 OUTPUT 717;":SENSe:TR1:NARRow1:TIME:RESolution?"
30 ENTER 717;A
```

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.SPAN

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.SPAN = *Value*

Value = SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.SPAN

Description

This command sets/gets time span, for the selected narrow *Nr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Time span
Data Type	Double precision floating point type (Double)
Range	0 ~ 1.00007936k
Preset Value	100m
Unit	s
Resolution	1n
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SENSe.TR1.NARRow1.TIME.SPAN = Var
Var = SCPI.SENSe.TR1.NARRow1.TIME.SPAN
```

Related Objects

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.DSTart

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.OFFSet

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.REFerence

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.RESolution

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.ZOUT

Equivalent Key

TR Menu: **Time Offset** > **Narrow Span**

TR Menu: **Time Offset** > **Narrow2 Span**

TR Menu: **Span** > **Narrow Span**

TR Menu: **Span** > **Narrow2 Span**

Equivalent SCPI Command

Syntax

:SENSe:TR[1-1]:NARRow[1-2]:TIME:SPAN {0 ~ 1.00007936k}

:SENSe:TR[1-1]:NARRow[1-2]:TIME:SPAN?

Query Response

{0 ~ 1.00007936k} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:TR1:NARRow1:TIME:SPAN 0"
20 OUTPUT 717;":SENSe:TR1:NARRow1:TIME:SPAN?"
30 ENTER 717;A
```

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.ZOUT

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.ZOUT = *Value*

Value = SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.ZOUT

Description

This command sets/gets narrow display zoom out, for the selected narrow *Nr* of the selected channel *Ch*.

NOTE This command is available when narrow 1.

Variable

Parameter	*Value*
Description	Narrow display zoom out
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.SENSe.TR1.NARRow1.TIME.ZOUT = Var
Var = SCPI.SENSe.TR1.NARRow1.TIME.ZOUT
```

Related Objects

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.DSTart

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.OFFSet

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.REFerence

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.RESolution

SCPI.SENSe.TR(Ch).NARRow(Nr).TIME.SPAN

Equivalent Key

TR Menu: **Setup** > **Advanced Setup** > **Narrow Zoom Out**

Equivalent SCPI Command

Syntax

:SENSe:TR[1-1]:NARRow[1-2]:TIME:ZOUT {ON|OFF|1|0}

:SENSe:TR[1-1]:NARRow[1-2]:TIME:ZOUT?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:TR1:NARRow1:TIME:ZOUT 1"
20 OUTPUT 717;":SENSe:TR1:NARRow1:TIME:ZOUT?"
30 ENTER 717;A
```

SCPI.SENSe.TR(Ch).POWer.INPut.LEVel.MAXimum

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.TR(Ch).POWer.INPut.LEVel.MAXimum = *Value*

Value = SCPI.SENSe.TR(Ch).POWer.INPut.LEVel.MAXimum

Description

This command sets/gets max input level, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Max Input Level
Data Type	Double precision floating point type (Double)
Range	-45 ~ 30
Preset Value	0
Unit	dBm
Resolution	100m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SENSe.TR1.POWer.INPut.LEVel.MAXimum = Var
Var = SCPI.SENSe.TR1.POWer.INPut.LEVel.MAXimum
```

Equivalent Key

TR Menu: **Setup** > **Max Input Level**

Equivalent SCPI Command

Syntax

:SENSe:TR[1-1]:POWer:INPut:LEVel:MAXimum {-45 ~ 30}

:SENSe:TR[1-1]:POWer:INPut:LEVel:MAXimum?

Query Response

{-45 ~ 30} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:TR1:POWer:INPut:LEVel:MAXimum 0"
20 OUTPUT 717;":SENSe:TR1:POWer:INPut:LEVel:MAXimum ?"
30 ENTER 717;A
```

SCPI.SENSe.TR(Ch).WIDE.FREQuency.MAXimum

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.TR(Ch).WIDE.FREQuency.MAXimum = *Value*

Value = SCPI.SENSe.TR(Ch).WIDE.FREQuency.MAXimum

Description

This command sets/gets max frequency, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Max frequency
Data Type	Double precision floating point type (Double)
Range	150M ~ 337G
Preset Value	1.2G
Unit	Hz
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 150000000
SCPI.SENSe.TR1.WIDE.FREQuency.MAXimum = Var
Var = SCPI.SENSe.TR1.WIDE.FREQuency.MAXimum
```

Equivalent Key

TR Menu: **Setup** > **Wide Max Frequency**

Equivalent SCPI Command

Syntax

:SENSe:TR[1-1]:WIDE:FREQuency:MAXimum {150M ~ 337G}

:SENSe:TR[1-1]:WIDE:FREQuency:MAXimum?

Query Response

{150M ~ 337G} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:TR1:WIDE:FREQuency:MAXimum 150000000"
20 OUTPUT 717;":SENSe:TR1:WIDE:FREQuency:MAXimum ?"
30 ENTER 717;A
```

SCPI.SENSe.TR(Ch).WIDE.SWEep.POINts

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.SENSe.TR(Ch).WIDE.SWEep.POINts

Description

This command gets number of points, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Number of points
Data Type	Long integer type (Long)
Range	801 ~ 1251
Preset Value	801
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 801
Var = SCPI.SENSe.TR1.WIDE.SWEep.POINts
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SENSe:TR[1-1]:WIDE:SWEep:POINts?

Query Response

{801 ~ 1251} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:TR1:WIDE:SWEep:POINts ?"
20 ENTER 717;A
```

SCPI.SENSe.TR(Ch).WIDE.TIME.OFFSet

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.TR(Ch).WIDE.TIME.OFFSet = *Value*

Value = SCPI.SENSe.TR(Ch).WIDE.TIME.OFFSet

Description

This command sets/gets offset for reference point, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Offset for reference point
Data Type	Double precision floating point type (Double)
Range	-8 ~ 11
Preset Value	0
Unit	s
Resolution	10n
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SENSe.TR1.WIDE.TIME.OFFSet = Var
Var = SCPI.SENSe.TR1.WIDE.TIME.OFFSet
```

Related Objects

SCPI.SENSe.TR(Ch).WIDE.TIME.OFFSet

SCPI.SENSe.TR(Ch).WIDE.TIME.REFerence

SCPI.SENSe.TR(Ch).WIDE.TIME.SPAN

Equivalent Key

TR Menu: **Span** > **Wide Time Offset**

Equivalent SCPI Command

Syntax

:SENSe:TR[1-1]:WIDE:TIME:OFFSet {-8 ~ 11}

:SENSe:TR[1-1]:WIDE:TIME:OFFSet?

Query Response

{-8 ~ 11} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:TR1:WIDE:TIME:OFFSet 0"
20 OUTPUT 717;":SENSe:TR1:WIDE:TIME:OFFSet ?"
30 ENTER 717;A
```

SCPI.SENSe.TR(Ch).WIDE.TIME.REFerence

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.TR(Ch).WIDE.TIME.REFerence = *Value*

Value = SCPI.SENSe.TR(Ch).WIDE.TIME.REFerence

Description

This command sets/gets reference position for span, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Reference position for span
Data Type	Character string type (String)
Range	LEFT|CENTer|RIGHt
Preset Value	CENTer
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "LEFT"
SCPI.SENSe.TR1.WIDE.TIME.REFerence = Var
Var = SCPI.SENSe.TR1.WIDE.TIME.REFerence
```

Related Objects

SCPI.SENSe.TR(Ch).WIDE.TIME.OFFSet

SCPI.SENSe.TR(Ch).WIDE.TIME.SPAN

Equivalent Key

TR Menu: **Span** > **Wide Ref Position**

Equivalent SCPI Command

Syntax

:SENSe:TR[1-1]:WIDE:TIME:REFerence {LEFT|CENTer|RIGHt}

:SENSe:TR[1-1]:WIDE:TIME:REFerence?

Query Response

{LEFT|CENTer|RIGHt} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:TR1:WIDE:TIME:REFerence LEFT"
20 OUTPUT 717;":SENSe:TR1:WIDE:TIME:REFerence ?"
30 ENTER 717;A$
```

SCPI.SENSe.TR(Ch).WIDE.TIME.SPAN

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.TR(Ch).WIDE.TIME.SPAN = *Value*

Value = SCPI.SENSe.TR(Ch).WIDE.TIME.SPAN

Description

This command sets/gets time span, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Time span
Data Type	Double precision floating point type (Double)
Range	0 ~ 10
Preset Value	100m
Unit	s
Resolution	10n
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SENSe.TR1.WIDE.TIME.SPAN = Var
Var = SCPI.SENSe.TR1.WIDE.TIME.SPAN
```

Related Objects

SCPI.SENSe.TR(Ch).WIDE.TIME.OFFSet

SCPI.SENSe.TR(Ch).WIDE.TIME.REFerence

Equivalent Key

TR Menu: **Span** > **Wide Span**

Equivalent SCPI Command

Syntax

:SENSe:TR[1-1]:WIDE:TIME:SPAN {0 ~ 10}

:SENSe:TR[1-1]:WIDE:TIME:SPAN?

Query Response

{0 ~ 10} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:TR1:WIDE:TIME:SPAN 0"
20 OUTPUT 717;":SENSe:TR1:WIDE:TIME:SPAN ?"
30 ENTER 717;A
```

SCPI.SENSe.UDConverter.HARMonic

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.UDConverter.HARMonic = *Value*

Value = SCPI.SENSe.UDConverter.HARMonic

Description

This command sets/gets user downconverter harmonic of LO for mixing.

Variable

Parameter	*Value*
Description	User downconverter harmonic of LO for mixing
Data Type	Long integer type (Long)
Range	1 ~ 34
Preset Value	1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.SENSe.UDConverter.HARMonic = Var
Var = SCPI.SENSe.UDConverter.HARMonic
```

Related Objects

SCPI.SENSe.UDConverter.LO

SCPI.SENSe.UDConverter.MODE

SCPI.SENSe.UDConverter.STATe

Equivalent Key

PN Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **Harmonic #**

SP Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **Harmonic #**

FP Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **Harmonic #**

TR Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **Harmonic #**

AM Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **Harmonic #**

BB Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **Harmonic #**

USER Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **Harmonic #**

Equivalent SCPI Command

Syntax

:SENSe:UDConverter:HARMonic {1 ~ 34}

:SENSe:UDConverter:HARMonic?

Query Response

{1 ~ 34} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:UDConverter:HARMonic 1 "
20 OUTPUT 717;":SENSe:UDConverter:HARMonic ?"
30 ENTER 717;A
```

SCPI.SENSe.UDConverter.LO

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.UDConverter.LO = *Value*

Value = SCPI.SENSe.UDConverter.LO

Description

This command sets/gets user downconverter local frequency.

Variable

Parameter	*Value*
Description	User downconverter local frequency
Data Type	Double precision floating point type (Double)
Range	10M ~ 330G
Preset Value	3G
Unit	Hz
Resolution	100m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 10000000
SCPI.SENSe.UDConverter.LO = Var
Var = SCPI.SENSe.UDConverter.LO
```

Related Objects

SCPI.SENSe.UDConverter.HARMonic

SCPI.SENSe.UDConverter.MODE

SCPI.SENSe.UDConverter.STATe

Equivalent Key

PN Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **LO Frequency**

SP Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **LO Frequency**

FP Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **LO Frequency**

TR Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **LO Frequency**

AM Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **LO Frequency**

BB Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **LO Frequency**

USER Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **LO Frequency**

Equivalent SCPI Command

Syntax

:SENSe:UDConverter:LO {10M ~ 330G}

:SENSe:UDConverter:LO?

Query Response

{10M ~ 330G} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:UDConverter:LO 10000000"
20 OUTPUT 717;":SENSe:UDConverter:LO ?"
30 ENTER 717;A
```

SCPI.SENSe.UDConverter.MODE

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.UDConverter.MODE = *Value*

Value = SCPI.SENSe.UDConverter.MODE

Description

This command sets/gets user downconverter mixer mode.

Variable

Parameter	*Value*
Description	User downconverter mixer mode
Data Type	Character string type (String)
Range	USB|LSB
Preset Value	USB
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "USB"
SCPI.SENSe.UDConverter.MODE = Var
Var = SCPI.SENSe.UDConverter.MODE
```

Related Objects

SCPI.SENSe.UDConverter.HARMonic

SCPI.SENSe.UDConverter.LO

SCPI.SENSe.UDConverter.STATe

Equivalent Key

PN Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **Conversion Mode**

SP Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **Conversion Mode**

FP Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **Conversion Mode**

TR Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **Conversion Mode**

AM Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **Conversion Mode**

BB Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **Conversion Mode**

USER Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **Conversion Mode**

Equivalent SCPI Command

Syntax

:SENSe:UDConverter:MODE {USB|LSB}

:SENSe:UDConverter:MODE?

Query Response

{USB|LSB} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:UDConverter:MODE USB"
20 OUTPUT 717;":SENSe:UDConverter:MODE ?"
30 ENTER 717;A$
```

SCPI.SENSe.UDConverter.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.SENSe.UDConverter.STATe = *Value*

Value = SCPI.SENSe.UDConverter.STATe

Description

This command sets/gets user downconverter ON/OFF.

Variable

Parameter	*Value*
Description	User downconverter (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.SENSe.UDConverter.STATe = Var
Var = SCPI.SENSe.UDConverter.STATe
```

Related Objects

SCPI.SENSe.UDConverter.HARMonic

SCPI.SENSe.UDConverter.LO

SCPI.SENSe.UDConverter.MODE

Equivalent Key

PN Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **Frequency Offset**

SP Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **Frequency Offset**

FP Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **Frequency Offset**

TR Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **Frequency Offset**

AM Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **Frequency Offset**

BB Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **Frequency Offset**

USER Menu: **System** > **Instrument Setup** > **Frequency Offset (User Downconv.)** > **Frequency Offset**

Equivalent SCPI Command

Syntax

:SENSe:UDConverter:STATe {ON|OFF|1|0}

:SENSe:UDConverter:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SENSe:UDConverter:STATe 1"
20 OUTPUT 717;":SENSe:UDConverter:STATe ?"
30 ENTER 717;A
```

Source

SCPI.SOURce.FP(Ch).SWEep.PARameter

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.FP(Ch).SWEep.PARameter = *Value*

Value = SCPI.SOURce.FP(Ch).SWEep.PARameter

Description

This command sets/gets sweep parameter, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Sweep parameter
Data Type	Character string type (String)
Range	CONTrol|POWer
Preset Value	CONTrol
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "CONTrol"
SCPI.SOURce.FP1.SWEep.PARameter = Var
Var = SCPI.SOURce.FP1.SWEep.PARameter
```

Related Objects

SCPI.SOURce.FP(Ch).SWEep.POINts

Equivalent Key

FP Menu: **Setup** > **Sweep Parameter**

Equivalent SCPI Command

Syntax

:SOURce:FP[1-1]:SWEep:PARameter {CONTrol|POWer}

:SOURce:FP[1-1]:SWEep:PARameter?

Query Response

{CONTrol|POWer} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:FP1:SWEep:PARameter CONTrol"
20 OUTPUT 717;":SOURce:FP1:SWEep:PARameter ?"
30 ENTER 717;A$
```

SCPI.SOURce.FP(Ch).SWEep.POINts

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.FP(Ch).SWEep.POINts = *Value*

Value = SCPI.SOURce.FP(Ch).SWEep.POINts

Description

This command sets/gets number of points, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Number of points
Data Type	Long integer type (Long)
Range	2 ~ 1001
Preset Value	201
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Long
Var= 2
SCPI.SOURce.FP1.SWEep.POINts = Var
Var = SCPI.SOURce.FP1.SWEep.POINts
```

Related Objects

SCPI.SOURce.FP(Ch).SWEep.PARameter

Equivalent Key

FP Menu: **Setup** > **Points**

Equivalent SCPI Command

Syntax

:SOURce:FP[1-1]:SWEep:POINts {2 ~ 1001}

:SOURce:FP[1-1]:SWEep:POINts?

Query Response

{2 ~ 1001} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:FP1:SWEep:POINts 2 "
20 OUTPUT 717;":SOURce:FP1:SWEep:POINts ?"
30 ENTER 717;A
```

SCPI.SOURce.FP(Ch).VOLTage.CONTrol.CENTer

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.FP(Ch).VOLTage.CONTrol.CENTer = *Value*

Value = SCPI.SOURce.FP(Ch).VOLTage.CONTrol.CENTer

Description

This command sets/gets voltage control center, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Voltage control center
Data Type	Double precision floating point type (Double)
Range	-15 ~ 35
Preset Value	50u
Unit	V
Resolution	50u
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SOURce.FP1.VOLTage.CONTrol.CENTer = Var
Var = SCPI.SOURce.FP1.VOLTage.CONTrol.CENTer
```

Related Objects

SCPI.SOURce.FP(Ch).VOLTage.CONTrol.SPAN

SCPI.SOURce.FP(Ch).VOLTage.CONTrol.STARt

SCPI.SOURce.FP(Ch).VOLTage.CONTrol.STOP

Equivalent Key

FP Menu: **Start/Center** > **DC Control Center**

FP Menu: **Stop/Span** > **DC Control Center**

Equivalent SCPI Command

Syntax

:SOURce:FP[1-1]:VOLTage:CONTrol:CENTer {-15 ~ 35}

:SOURce:FP[1-1]:VOLTage:CONTrol:CENTer?

Query Response

{-15 ~ 35} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:FP1:VOLTage:CONTrol:CENTer 0"
20 OUTPUT 717;":SOURce:FP1:VOLTage:CONTrol:CENTer ?"
30 ENTER 717;A
```

SCPI.SOURce.FP(Ch).VOLTage.CONTrol.SPAN

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.FP(Ch).VOLTage.CONTrol.SPAN = *Value*

Value = SCPI.SOURce.FP(Ch).VOLTage.CONTrol.SPAN

Description

This command sets/gets voltage control span, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Voltage control span
Data Type	Double precision floating point type (Double)
Range	0 ~ 50
Preset Value	100u
Unit	V
Resolution	100u
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SOURce.FP1.VOLTage.CONTrol.SPAN = Var
Var = SCPI.SOURce.FP1.VOLTage.CONTrol.SPAN
```

Related Objects

SCPI.SOURce.FP(Ch).VOLTage.CONTrol.CENTer

SCPI.SOURce.FP(Ch).VOLTage.CONTrol.STARt

SCPI.SOURce.FP(Ch).VOLTage.CONTrol.STOP

Equivalent Key

FP Menu: **Start/Center** > **DC Control Span**

FP Menu: **Stop/Span** > **DC Control Span**

Equivalent SCPI Command

Syntax

:SOURce:FP[1-1]:VOLTage:CONTrol:SPAN {0 ~ 50}

:SOURce:FP[1-1]:VOLTage:CONTrol:SPAN?

Query Response

{0 ~ 50} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:FP1:VOLTage:CONTrol:SPAN 0"
20 OUTPUT 717;":SOURce:FP1:VOLTage:CONTrol:SPAN ?"
30 ENTER 717;A
```

SCPI.SOURce.FP(Ch).VOLTage.CONTrol.STARt

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.FP(Ch).VOLTage.CONTrol.STARt = *Value*

Value = SCPI.SOURce.FP(Ch).VOLTage.CONTrol.STARt

Description

This command sets/gets voltage control start, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Voltage control start
Data Type	Double precision floating point type (Double)
Range	-15 ~ 35
Preset Value	0
Unit	V
Resolution	100u
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SOURce.FP1.VOLTage.CONTrol.STARt = Var
Var = SCPI.SOURce.FP1.VOLTage.CONTrol.STARt
```

Related Objects

SCPI.SOURce.FP(Ch).VOLTage.CONTrol.CENTer

SCPI.SOURce.FP(Ch).VOLTage.CONTrol.SPAN

SCPI.SOURce.FP(Ch).VOLTage.CONTrol.STOP

Equivalent Key

FP Menu: **Start/Center** > **DC Control Start**

FP Menu: **Stop/Span** > **DC Control Start**

Equivalent SCPI Command

Syntax

:SOURce:FP[1-1]:VOLTage:CONTrol:STARt {-15 ~ 35}

:SOURce:FP[1-1]:VOLTage:CONTrol:STARt?

Query Response

{-15 ~ 35} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:FP1:VOLTage:CONTrol:STARt 0"
20 OUTPUT 717;":SOURce:FP1:VOLTage:CONTrol:STARt ?"
30 ENTER 717;A
```

SCPI.SOURce.FP(Ch).VOLTage.CONTrol.STOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.FP(Ch).VOLTage.CONTrol.STOP = *Value*

Value = SCPI.SOURce.FP(Ch).VOLTage.CONTrol.STOP

Description

This command sets/gets voltage control stop, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Voltage control stop
Data Type	Double precision floating point type (Double)
Range	-15 ~ 35
Preset Value	100u
Unit	V
Resolution	100u
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SOURce.FP1.VOLTage.CONTrol.STOP = Var
Var = SCPI.SOURce.FP1.VOLTage.CONTrol.STOP
```

Related Objects

SCPI.SOURce.FP(Ch).VOLTage.CONTrol.CENTer

SCPI.SOURce.FP(Ch).VOLTage.CONTrol.SPAN

SCPI.SOURce.FP(Ch).VOLTage.CONTrol.STARt

Equivalent Key

FP Menu: **Start/Center** > **DC Control Stop**

FP Menu: **Stop/Span** > **DC Control Stop**

Equivalent SCPI Command

Syntax

:SOURce:FP[1-1]:VOLTage:CONTrol:STOP {-15 ~ 35}

:SOURce:FP[1-1]:VOLTage:CONTrol:STOP?

Query Response

{-15 ~ 35} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:FP1:VOLTage:CONTrol:STOP 0"
20 OUTPUT 717;":SOURce:FP1:VOLTage:CONTrol:STOP ?"
30 ENTER 717;A
```

SCPI.SOURce.FP(Ch).VOLTage.POWer.CENTer

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.FP(Ch).VOLTage.POWer.CENTer = *Value*

Value = SCPI.SOURce.FP(Ch).VOLTage.POWer.CENTer

Description

This command sets/gets voltage power center, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Voltage power center
Data Type	Double precision floating point type (Double)
Range	0 ~ 16
Preset Value	500u
Unit	V
Resolution	500u
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SOURce.FP1.VOLTage.POWer.CENTer = Var
Var = SCPI.SOURce.FP1.VOLTage.POWer.CENTer
```

Related Objects

SCPI.SOURce.FP(Ch).VOLTage.POWer.SPAN

SCPI.SOURce.FP(Ch).VOLTage.POWer.STARt

SCPI.SOURce.FP(Ch).VOLTage.POWer.STOP

Equivalent Key

FP Menu: **Start/Center** > **DC Power Center**

FP Menu: **Stop/Span** > **DC Power Center**

Equivalent SCPI Command

Syntax

:SOURce:FP[1-1]:VOLTage:POWer:CENTer {0 ~ 16}

:SOURce:FP[1-1]:VOLTage:POWer:CENTer?

Query Response

{0 ~ 16} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:FP1:VOLTage:POWer:CENTer 0"
20 OUTPUT 717;":SOURce:FP1:VOLTage:POWer:CENTer ?"
30 ENTER 717;A
```

SCPI.SOURce.FP(Ch).VOLTage.POWer.SPAN

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.FP(Ch).VOLTage.POWer.SPAN = *Value*

Value = SCPI.SOURce.FP(Ch).VOLTage.POWer.SPAN

Description

This command sets/gets voltage power span, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Voltage power span
Data Type	Double precision floating point type (Double)
Range	0 ~ 16
Preset Value	1m
Unit	V
Resolution	1m
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SOURce.FP1.VOLTage.POWer.SPAN = Var
Var = SCPI.SOURce.FP1.VOLTage.POWer.SPAN
```

Related Objects

SCPI.SOURce.FP(Ch).VOLTage.POWer.CENTer

SCPI.SOURce.FP(Ch).VOLTage.POWer.STARt

SCPI.SOURce.FP(Ch).VOLTage.POWer.STOP

Equivalent Key

FP Menu: **Start/Center** > **DC Power Span**

FP Menu: **Stop/Span** > **DC Power Span**

Equivalent SCPI Command

Syntax

:SOURce:FP[1-1]:VOLTage:POWer:SPAN {0 ~ 16}

:SOURce:FP[1-1]:VOLTage:POWer:SPAN?

Query Response

{0 ~ 16} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:FP1:VOLTage:POWer:SPAN 0"
20 OUTPUT 717;":SOURce:FP1:VOLTage:POWer:SPAN ?"
30 ENTER 717;A
```

SCPI.SOURce.FP(Ch).VOLTage.POWer.STARt

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.FP(Ch).VOLTage.POWer.STARt = *Value*

Value = SCPI.SOURce.FP(Ch).VOLTage.POWer.STARt

Description

This command sets/gets voltage power start, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Voltage power start
Data Type	Double precision floating point type (Double)
Range	0 ~ 16
Preset Value	0
Unit	V
Resolution	1m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SOURce.FP1.VOLTage.POWer.STARt = Var
Var = SCPI.SOURce.FP1.VOLTage.POWer.STARt
```

Related Objects

SCPI.SOURce.FP(Ch).VOLTage.POWer.CENTer

SCPI.SOURce.FP(Ch).VOLTage.POWer.SPAN

SCPI.SOURce.FP(Ch).VOLTage.POWer.STOP

Equivalent Key

FP Menu: **Start/Center** > **DC Power Start**

FP Menu: **Stop/Span** > **DC Power Start**

Equivalent SCPI Command

Syntax

:SOURce:FP[1-1]:VOLTage:POWer:STARt {0 ~ 16}

:SOURce:FP[1-1]:VOLTage:POWer:STARt?

Query Response

{0 ~ 16} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:FP1:VOLTage:POWer:STARt 0"
20 OUTPUT 717;":SOURce:FP1:VOLTage:POWer:STARt ?"
30 ENTER 717;A
```

SCPI.SOURce.FP(Ch).VOLTage.POWer.STOP

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.FP(Ch).VOLTage.POWer.STOP = *Value*

Value = SCPI.SOURce.FP(Ch).VOLTage.POWer.STOP

Description

This command sets/gets voltage power stop, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Voltage power stop
Data Type	Double precision floating point type (Double)
Range	0 ~ 16
Preset Value	1m
Unit	V
Resolution	1m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SOURce.FP1.VOLTage.POWer.STOP = Var
Var = SCPI.SOURce.FP1.VOLTage.POWer.STOP
```

Related Objects

SCPI.SOURce.FP(Ch).VOLTage.POWer.CENTer

SCPI.SOURce.FP(Ch).VOLTage.POWer.SPAN

SCPI.SOURce.FP(Ch).VOLTage.POWer.STARt

Equivalent Key

FP Menu: **Start/Center** > **DC Power Stop**

FP Menu: **Stop/Span** > **DC Power Stop**

Equivalent SCPI Command

Syntax

:SOURce:FP[1-1]:VOLTage:POWer:STOP {0 ~ 16}

:SOURce:FP[1-1]:VOLTage:POWer:STOP?

Query Response

{0 ~ 16} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:FP1:VOLTage:POWer:STOP 0"
20 OUTPUT 717;":SOURce:FP1:VOLTage:POWer:STOP ?"
30 ENTER 717;A
```

SCPI.SOURce.VOLTage.CONTrol.AFC.FBANd

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.VOLTage.CONTrol.AFC.FBANd = *Value*

Value = SCPI.SOURce.VOLTage.CONTrol.AFC.FBANd

Description

This command sets/gets frequency band.

Variable

Parameter	*Value*
Description	Frequency band
Data Type	Character string type (String)
Range	LOW|HIGH|BAND3|BAND4
Preset Value	HIGH
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "LOW"
SCPI.SOURce.VOLTage.CONTrol.AFC.FBANd = Var
Var = SCPI.SOURce.VOLTage.CONTrol.AFC.FBANd
```

Related Objects

SCPI.SOURce.VOLTage.CONTrol.AFC.IMMediate

SCPI.SOURce.VOLTage.CONTrol.AFC.INPut.LEVel.MAXimum

SCPI.SOURce.VOLTage.CONTrol.AFC.ITERation

SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.HIGH

SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.LOW

SCPI.SOURce.VOLTage.CONTrol.AFC.SENSitivity

SCPI.SOURce.VOLTage.CONTrol.AFC.STATe

SCPI.SOURce.VOLTage.CONTrol.AFC.TARGet

SCPI.SOURce.VOLTage.CONTrol.AFC.TOLerance

Equivalent Key

PN Menu: **DC Control Voltage** > **Auto Freq Control** > **Frequency Band**

SP Menu: **DC Control Voltage** > **Auto Freq Control** > **Frequency Band**

FP Menu: **DC Control Voltage** > **Auto Freq Control** > **Frequency Band**

TR Menu: **DC Control Voltage** > **Auto Freq Control** > **Frequency Band**

AM Menu: **DC Control Voltage** > **Auto Freq Control** > **Frequency Band**

BB Menu: **DC Control Voltage** > **Auto Freq Control** > **Frequency Band**

USER Menu: **DC Control Voltage** > **Auto Freq Control** > **Frequency Band**

Equivalent SCPI Command

Syntax

:SOURce:VOLTage:CONTrol:AFC:FBANd {LOW|HIGH|BAND3|BAND4}

:SOURce:VOLTage:CONTrol:AFC:FBANd?

Query Response

{LOW|HIGH|BAND3|BAND4} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:VOLTage:CONTrol:AFC:FBANd LOW"
20 OUTPUT 717;":SOURce:VOLTage:CONTrol:AFC:FBANd ?"
30 ENTER 717;A$
```

SCPI.SOURce.VOLTage.CONTrol.AFC.IMMediate

Object Type

Method (**Write Only**)

Syntax

SCPI.SOURce.VOLTage.CONTrol.AFC.IMMediate

Description

This command immediately executes AFC .

Examples

SCPI.SOURce.VOLTage.CONTrol.AFC.IMMediate

Related Objects

SCPI.SOURce.VOLTage.CONTrol.AFC.FBANd

SCPI.SOURce.VOLTage.CONTrol.AFC.INPut.LEVel.MAXimum

SCPI.SOURce.VOLTage.CONTrol.AFC.ITERation

SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.HIGH

SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.LOW

SCPI.SOURce.VOLTage.CONTrol.AFC.SENSitivity

SCPI.SOURce.VOLTage.CONTrol.AFC.STATe

SCPI.SOURce.VOLTage.CONTrol.AFC.TARGet

SCPI.SOURce.VOLTage.CONTrol.AFC.TOLerance

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SOURce:VOLTage:CONTrol:AFC:IMMediate

Example of use

```
10 OUTPUT 717;":SOURce:VOLTage:CONTrol:AFC:IMMediate"
```

SCPI.SOURce.VOLTage.CONTrol.AFC.INPut.LEVel.MAXimum

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.VOLTage.CONTrol.AFC.INPut.LEVel.MAXimum = *Value*

Value = SCPI.SOURce.VOLTage.CONTrol.AFC.INPut.LEVel.MAXimum

Description

This command sets/gets max input level.

Variable

Parameter	*Value*
Description	Max Input Level
Data Type	Double precision floating point type (Double)
Range	-45 ~ 30
Preset Value	0
Unit	dBm
Resolution	100m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SOURce.VOLTage.CONTrol.AFC.INPut.LEVel.MAXimum = Var
Var = SCPI.SOURce.VOLTage.CONTrol.AFC.INPut.LEVel.MAXimum
```

Related Objects

SCPI.SOURce.VOLTage.CONTrol.AFC.FBANd

SCPI.SOURce.VOLTage.CONTrol.AFC.IMMediate

SCPI.SOURce.VOLTage.CONTrol.AFC.ITERation

SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.HIGH

SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.LOW

SCPI.SOURce.VOLTage.CONTrol.AFC.SENSitivity

SCPI.SOURce.VOLTage.CONTrol.AFC.STATe

SCPI.SOURce.VOLTage.CONTrol.AFC.TARGet

SCPI.SOURce.VOLTage.CONTrol.AFC.TOLerance

Equivalent Key

PN Menu: **DC Control Voltage** > **Auto Freq Control** > **Max Input Level**

SP Menu: **DC Control Voltage** > **Auto Freq Control** > **Max Input Level**

FP Menu: **DC Control Voltage** > **Auto Freq Control** > **Max Input Level**

TR Menu: **DC Control Voltage** > **Auto Freq Control** > **Max Input Level**

AM Menu: **DC Control Voltage** > **Auto Freq Control** > **Max Input Level**

BB Menu: **DC Control Voltage** > **Auto Freq Control** > **Max Input Level**

USER Menu: **DC Control Voltage** > **Auto Freq Control** > **Max Input Level**

Equivalent SCPI Command

Syntax

:SOURce:VOLTage:CONTrol:AFC:INPut:LEVel:MAXimum {-45 ~ 30}

:SOURce:VOLTage:CONTrol:AFC:INPut:LEVel:MAXimum?

Query Response

{-45 ~ 30} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:VOLTage:CONTrol:AFC:INPut:LEVel:MAXimum 0"
20 OUTPUT 717;":SOURce:VOLTage:CONTrol:AFC:INPut:LEVel:MAXimum ?"
30 ENTER 717;A
```

SCPI.SOURce.VOLTage.CONTrol.AFC.ITERation

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.VOLTage.CONTrol.AFC.ITERation = *Value*

Value = SCPI.SOURce.VOLTage.CONTrol.AFC.ITERation

Description

This command sets/gets the maximum number of times that the measurement and calculation are repeated in AFC.

Variable

Parameter	*Value*
Description	The maximum number of times that the measurement and calculation are repeated in AFC
Data Type	Long integer type (Long)
Range	1 ~ 99
Preset Value	10
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Long
Var= 1
SCPI.SOURce.VOLTage.CONTrol.AFC.ITERation = Var
Var = SCPI.SOURce.VOLTage.CONTrol.AFC.ITERation
```

Related Objects

SCPI.SOURce.VOLTage.CONTrol.AFC.FBANd

SCPI.SOURce.VOLTage.CONTrol.AFC.IMMediate

SCPI.SOURce.VOLTage.CONTrol.AFC.INPut.LEVel.MAXimum

SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.HIGH

SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.LOW

SCPI.SOURce.VOLTage.CONTrol.AFC.SENSitivity

SCPI.SOURce.VOLTage.CONTrol.AFC.STATe

SCPI.SOURce.VOLTage.CONTrol.AFC.TARGet

SCPI.SOURce.VOLTage.CONTrol.AFC.TOLerance

Equivalent Key

PN Menu: **DC Control Voltage** > **Auto Freq Control** > **Max Iteration**

SP Menu: **DC Control Voltage** > **Auto Freq Control** > **Max Iteration**

FP Menu: **DC Control Voltage** > **Auto Freq Control** > **Max Iteration**

TR Menu: **DC Control Voltage** > **Auto Freq Control** > **Max Iteration**

AM Menu: **DC Control Voltage** > **Auto Freq Control** > **Max Iteration**

BB Menu: **DC Control Voltage** > **Auto Freq Control** > **Max Iteration**

USER Menu: **DC Control Voltage** > **Auto Freq Control** > **Max Iteration**

Equivalent SCPI Command

Syntax

```
:SOURce:VOLTage:CONTrol:AFC:ITERation {1 ~ 99}
:SOURce:VOLTage:CONTrol:AFC:ITERation?
```

Query Response

{1 ~ 99} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:VOLTage:CONTrol:AFC:ITERation 1 "
20 OUTPUT 717;":SOURce:VOLTage:CONTrol:AFC:ITERation ?"
30 ENTER 717;A
```

SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.HIGH

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.HIGH = *Value*

Value = SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.HIGH

Description

This command sets/gets AFC voltage control high limit.

Variable

Parameter	*Value*
Description	AFC Voltage control high limit
Data Type	Double precision floating point type (Double)
Range	-15 ~ 35
Preset Value	35
Unit	V
Resolution	100u
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.HIGH = Var
Var = SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.HIGH
```

Related Objects

SCPI.SOURce.VOLTage.CONTrol.AFC.FBANd

SCPI.SOURce.VOLTage.CONTrol.AFC.IMMediate

SCPI.SOURce.VOLTage.CONTrol.AFC.INPut.LEVel.MAXimum

SCPI.SOURce.VOLTage.CONTrol.AFC.ITERation

SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.LOW

SCPI.SOURce.VOLTage.CONTrol.AFC.SENSitivity

SCPI.SOURce.VOLTage.CONTrol.AFC.STATe

SCPI.SOURce.VOLTage.CONTrol.AFC.TARGet

SCPI.SOURce.VOLTage.CONTrol.AFC.TOLerance

Equivalent Key

PN Menu: **DC Control Voltage** > **Auto Freq Control** > **Max Ctrl Voltage Limit**

SP Menu: **DC Control Voltage** > **Auto Freq Control** > **Max Ctrl Voltage Limit**

FP Menu: **DC Control Voltage** > **Auto Freq Control** > **Max Ctrl Voltage Limit**

TR Menu: **DC Control Voltage** > **Auto Freq Control** > **Max Ctrl Voltage Limit**

AM Menu: **DC Control Voltage** > **Auto Freq Control** > **Max Ctrl Voltage Limit**

BB Menu: **DC Control Voltage** > **Auto Freq Control** > **Max Ctrl Voltage Limit**

USER Menu: **DC Control Voltage** > **Auto Freq Control** > **Max Ctrl Voltage Limit**

Equivalent SCPI Command

Syntax

:SOURce:VOLTage:CONTrol:AFC:LIMit:HIGH {-15 ~ 35}

:SOURce:VOLTage:CONTrol:AFC:LIMit:HIGH?

Query Response

{-15 ~ 35} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:VOLTage:CONTrol:AFC:LIMit:HIGH 0"
20 OUTPUT 717;":SOURce:VOLTage:CONTrol:AFC:LIMit:HIGH ?"
30 ENTER 717;A
```

SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.LOW

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.LOW = *Value*

Value = SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.LOW

Description

This command sets/gets AFC voltage control low limit.

Variable

Parameter	*Value*
Description	AFC Voltage control low limit
Data Type	Double precision floating point type (Double)
Range	-15 ~ 35
Preset Value	-15
Unit	V
Resolution	100u
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.LOW = Var
Var = SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.LOW
```

Related Objects

SCPI.SOURce.VOLTage.CONTrol.AFC.FBANd

SCPI.SOURce.VOLTage.CONTrol.AFC.IMMediate

SCPI.SOURce.VOLTage.CONTrol.AFC.INPut.LEVel.MAXimum

SCPI.SOURce.VOLTage.CONTrol.AFC.ITERation

SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.HIGH

SCPI.SOURce.VOLTage.CONTrol.AFC.SENSitivity

SCPI.SOURce.VOLTage.CONTrol.AFC.STATe

SCPI.SOURce.VOLTage.CONTrol.AFC.TARGet

SCPI.SOURce.VOLTage.CONTrol.AFC.TOLerance

Equivalent Key

PN Menu: **DC Control Voltage** > **Auto Freq Control** > **Min Ctrl Voltage Limit**

SP Menu: **DC Control Voltage** > **Auto Freq Control** > **Min Ctrl Voltage Limit**

FP Menu: **DC Control Voltage** > **Auto Freq Control** > **Min Ctrl Voltage Limit**

TR Menu: **DC Control Voltage** > **Auto Freq Control** > **Min Ctrl Voltage Limit**

AM Menu: **DC Control Voltage** > **Auto Freq Control** > **Min Ctrl Voltage Limit**

BB Menu: **DC Control Voltage** > **Auto Freq Control** > **Min Ctrl Voltage Limit**

USER Menu: **DC Control Voltage** > **Auto Freq Control** > **Min Ctrl Voltage Limit**

Equivalent SCPI Command

Syntax

```
:SOURce:VOLTage:CONTrol:AFC:LIMit:LOW {-15 ~ 35}
:SOURce:VOLTage:CONTrol:AFC:LIMit:LOW?
```

Query Response

```
{-15 ~ 35} <newline><^END>
```

Example of use

```
10 OUTPUT 717;":SOURce:VOLTage:CONTrol:AFC:LIMit:LOW 0"
20 OUTPUT 717;":SOURce:VOLTage:CONTrol:AFC:LIMit:LOW ?"
30 ENTER 717;A
```

SCPI.SOURce.VOLTage.CONTrol.AFC.SENSitivity

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.VOLTage.CONTrol.AFC.SENSitivity = *Value*

Value = SCPI.SOURce.VOLTage.CONTrol.AFC.SENSitivity

Description

This command sets/gets the DUT's approximate tuning sensitivity used in AFC.

Variable

Parameter	*Value*
Description	DUT's approximate tuning sensitivity used in AFC
Data Type	Double precision floating point type (Double)
Range	-5G ~ 5G
Preset Value	10M
Unit	Hz/V
Resolution	1
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= -5000000000
SCPI.SOURce.VOLTage.CONTrol.AFC.SENSitivity = Var
Var = SCPI.SOURce.VOLTage.CONTrol.AFC.SENSitivity
```

Related Objects

SCPI.SOURce.VOLTage.CONTrol.AFC.FBANd

SCPI.SOURce.VOLTage.CONTrol.AFC.IMMediate

SCPI.SOURce.VOLTage.CONTrol.AFC.INPut.LEVel.MAXimum

SCPI.SOURce.VOLTage.CONTrol.AFC.ITERation

SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.HIGH

SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.LOW

SCPI.SOURce.VOLTage.CONTrol.AFC.STATe

SCPI.SOURce.VOLTage.CONTrol.AFC.TARGet

SCPI.SOURce.VOLTage.CONTrol.AFC.TOLerance

Equivalent Key

PN Menu: **DC Control Voltage** > **Auto Freq Control** > **Sensitivity**

SP Menu: **DC Control Voltage** > **Auto Freq Control** > **Sensitivity**

FP Menu: **DC Control Voltage** > **Auto Freq Control** > **Sensitivity**

TR Menu: **DC Control Voltage** > **Auto Freq Control** > **Sensitivity**

AM Menu: **DC Control Voltage** > **Auto Freq Control** > **Sensitivity**

BB Menu: **DC Control Voltage** > **Auto Freq Control** > **Sensitivity**

USER Menu: **DC Control Voltage** > **Auto Freq Control** > **Sensitivity**

Equivalent SCPI Command

Syntax

:SOURce:VOLTage:CONTrol:AFC:SENSitivity {-5G ~ 5G}

:SOURce:VOLTage:CONTrol:AFC:SENSitivity?

Query Response

{-5G ~ 5G} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:VOLTage:CONTrol:AFC:SENSitivity -5000000000"
20 OUTPUT 717;":SOURce:VOLTage:CONTrol:AFC:SENSitivity ?"
30 ENTER 717;A
```

SCPI.SOURce.VOLTage.CONTrol.AFC.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.VOLTage.CONTrol.AFC.STATe = *Value*

Value = SCPI.SOURce.VOLTage.CONTrol.AFC.STATe

Description

This command sets/gets the AFC state for PN/SP.

Variable

Parameter	*Value*
Description	AFC state for PN/SP
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.SOURce.VOLTage.CONTrol.AFC.STATe = Var
Var = SCPI.SOURce.VOLTage.CONTrol.AFC.STATe
```

Related Objects

SCPI.SOURce.VOLTage.CONTrol.AFC.FBANd

SCPI.SOURce.VOLTage.CONTrol.AFC.IMMediate

SCPI.SOURce.VOLTage.CONTrol.AFC.INPut.LEVel.MAXimum

SCPI.SOURce.VOLTage.CONTrol.AFC.ITERation

SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.HIGH

SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.LOW

SCPI.SOURce.VOLTage.CONTrol.AFC.SENSitivity

SCPI.SOURce.VOLTage.CONTrol.AFC.TARGet

SCPI.SOURce.VOLTage.CONTrol.AFC.TOLerance

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SOURce:VOLTage:CONTrol:AFC[:STATe] {ON|OFF|1|0}

:SOURce:VOLTage:CONTrol:AFC[:STATe]?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:VOLTage:CONTrol:AFC[:STATe] 1"
20 OUTPUT 717;":SOURce:VOLTage:CONTrol:AFC[:STATe] ?"
30 ENTER 717;A
```

SCPI.SOURce.VOLTage.CONTrol.AFC.TARGet

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.VOLTage.CONTrol.AFC.TARGet = *Value*

Value = SCPI.SOURce.VOLTage.CONTrol.AFC.TARGet

Description

This command sets/gets the target frequency for AFC.

Variable

Parameter	*Value*
Description	Target frequency for AFC
Data Type	Double precision floating point type (Double)
Range	10M ~ 337G
Preset Value	1G
Unit	Hz
Resolution	100m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 10000000
SCPI.SOURce.VOLTage.CONTrol.AFC.TARGet = Var
Var = SCPI.SOURce.VOLTage.CONTrol.AFC.TARGet
```

Related Objects

SCPI.SOURce.VOLTage.CONTrol.AFC.FBANd

SCPI.SOURce.VOLTage.CONTrol.AFC.IMMediate

SCPI.SOURce.VOLTage.CONTrol.AFC.INPut.LEVel.MAXimum

SCPI.SOURce.VOLTage.CONTrol.AFC.ITERation

SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.HIGH

SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.LOW

SCPI.SOURce.VOLTage.CONTrol.AFC.SENSitivity

SCPI.SOURce.VOLTage.CONTrol.AFC.STATe

SCPI.SOURce.VOLTage.CONTrol.AFC.TOLerance

Equivalent Key

PN Menu: **DC Control Voltage** > **Auto Freq Control** > **Target**

SP Menu: **DC Control Voltage** > **Auto Freq Control** > **Target**

FP Menu: **DC Control Voltage** > **Auto Freq Control** > **Target**

TR Menu: **DC Control Voltage** > **Auto Freq Control** > **Target**

AM Menu: **DC Control Voltage** > **Auto Freq Control** > **Target**

BB Menu: **DC Control Voltage** > **Auto Freq Control** > **Target**

USER Menu: **DC Control Voltage** > **Auto Freq Control** > **Target**

Equivalent SCPI Command

Syntax

:SOURce:VOLTage:CONTrol:AFC:TARGet {10M ~ 337G}

:SOURce:VOLTage:CONTrol:AFC:TARGet?

Query Response

{10M ~ 337G} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:VOLTage:CONTrol:AFC:TARGet 10000000"
20 OUTPUT 717;":SOURce:VOLTage:CONTrol:AFC:TARGet ?"
30 ENTER 717;A
```

SCPI.SOURce.VOLTage.CONTrol.AFC.TOLerance

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.VOLTage.CONTrol.AFC.TOLerance = *Value*

Value = SCPI.SOURce.VOLTage.CONTrol.AFC.TOLerance

Description

This command sets/gets the allowable difference between the target frequency and the actual frequency settings in AFC.

Variable

Parameter	*Value*
Description	The allowable difference between the target frequency and the actual frequency settings in AFC
Data Type	Double precision floating point type (Double)
Range	20 ~ 10M
Preset Value	1k
Unit	Hz
Resolution	100m
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SOURce.VOLTage.CONTrol.AFC.TOLerance = Var
Var = SCPI.SOURce.VOLTage.CONTrol.AFC.TOLerance
```

Related Objects

SCPI.SOURce.VOLTage.CONTrol.AFC.FBANd

SCPI.SOURce.VOLTage.CONTrol.AFC.IMMediate

SCPI.SOURce.VOLTage.CONTrol.AFC.INPut.LEVel.MAXimum

SCPI.SOURce.VOLTage.CONTrol.AFC.ITERation

SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.HIGH

SCPI.SOURce.VOLTage.CONTrol.AFC.LIMit.LOW

SCPI.SOURce.VOLTage.CONTrol.AFC.SENSitivity

SCPI.SOURce.VOLTage.CONTrol.AFC.STATe

SCPI.SOURce.VOLTage.CONTrol.AFC.TARGet

Equivalent Key

PN Menu: **DC Control Voltage** > **Auto Freq Control** > **Tolerance**

SP Menu: **DC Control Voltage** > **Auto Freq Control** > **Tolerance**

FP Menu: **DC Control Voltage** > **Auto Freq Control** > **Tolerance**

TR Menu: **DC Control Voltage** > **Auto Freq Control** > **Tolerance**

AM Menu: **DC Control Voltage** > **Auto Freq Control** > **Tolerance**

BB Menu: **DC Control Voltage** > **Auto Freq Control** > **Tolerance**

USER Menu: **DC Control Voltage** > **Auto Freq Control** > **Tolerance**

Equivalent SCPI Command

Syntax

:SOURce:VOLTage:CONTrol:AFC:TOLerance {20 ~ 10M}

:SOURce:VOLTage:CONTrol:AFC:TOLerance?

Query Response

{20 ~ 10M} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:VOLTage:CONTrol:AFC:TOLerance 0"
20 OUTPUT 717;":SOURce:VOLTage:CONTrol:AFC:TOLerance ?"
30 ENTER 717;A
```

SCPI.SOURce.VOLTage.CONTrol.CORRection.COLLect.ACQuire

Object Type

Method (**Write Only**)

Syntax

SCPI.SOURce.VOLTage.CONTrol.CORRection.COLLect.ACQuire

Description

This command executes DC control drift calibration..

Examples

SCPI.SOURce.VOLTage.CONTrol.CORRection.COLLect.ACQuire

Related Objects

SCPI.SOURce.VOLTage.CONTrol.CORRection.STATe

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SOURce:VOLTage:CONTrol:CORRection:COLLect:ACQuire

Example of use

10 OUTPUT 717;":SOURce:VOLTage:CONTrol:CORRection:COLLect:ACQuire"

SCPI.SOURce.VOLTage.CONTrol.CORRection.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.VOLTage.CONTrol.CORRection.STATe = *Value*

Value = SCPI.SOURce.VOLTage.CONTrol.CORRection.STATe

Description

This command sets/gets DC control drift calibration state.

Variable

Parameter	*Value*
Description	DC control drift calibration state
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.SOURce.VOLTage.CONTrol.CORRection.STATe = Var
Var = SCPI.SOURce.VOLTage.CONTrol.CORRection.STATe
```

Related Objects

SCPI.SOURce.VOLTage.CONTrol.CORRection.COLLect.ACQuire

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SOURce:VOLTage:CONTrol:CORRection[:STATe] {ON|OFF|1|0}

:SOURce:VOLTage:CONTrol:CORRection[:STATe]?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:VOLTage:CONTrol:CORRection[:STATe] 1"
20 OUTPUT 717;":SOURce:VOLTage:CONTrol:CORRection[:STATe] ?"
30 ENTER 717;A
```

SCPI.SOURce.VOLTage.CONTrol.DELay

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.VOLTage.CONTrol.DELay = *Value*

Value = SCPI.SOURce.VOLTage.CONTrol.DELay

Description

This command sets/gets Source Control setting delay (in sec).

Variable

Parameter	*Value*
Description	Source Control setting delay
Data Type	Double precision floating point type (Double)
Range	0 ~ 1
Preset Value	100m
Unit	s
Resolution	1m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SOURce.VOLTage.CONTrol.DELay = Var
Var = SCPI.SOURce.VOLTage.CONTrol.DELay
```

Equivalent Key

PN Menu: **DC Control Voltage** > **DC Control Delay**

SP Menu: **DC Control Voltage** > **DC Control Delay**

FP Menu: **DC Control Voltage** > **DC Control Delay**

TR Menu: **DC Control Voltage** > **DC Control Delay**

AM Menu: **DC Control Voltage** > **DC Control Delay**

BB Menu: **DC Control Voltage** > **DC Control Delay**

USER Menu: **DC Control Voltage** > **DC Control Delay**

Equivalent SCPI Command

Syntax

:SOURce:VOLTage:CONTrol:DELay {0 ~ 1}

:SOURce:VOLTage:CONTrol:DELay?

Query Response

{0 ~ 1} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:VOLTage:CONTrol:DELay 0"
20 OUTPUT 717;":SOURce:VOLTage:CONTrol:DELay ?"
30 ENTER 717;A
```

SCPI.SOURce.VOLTage.CONTrol.LEVel.AMPLitude

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.VOLTage.CONTrol.LEVel.AMPLitude = *Value*

Value = SCPI.SOURce.VOLTage.CONTrol.LEVel.AMPLitude

Description

This command sets/gets fixed voltage control value at voltage power sweep.

Variable

Parameter	*Value*
Description	Fixed Voltage control value at Voltage power sweep
Data Type	Double precision floating point type (Double)
Range	-15 ~ 35
Preset Value	0
Unit	V
Resolution	100u
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SOURce.VOLTage.CONTrol.LEVel.AMPLitude = Var
Var = SCPI.SOURce.VOLTage.CONTrol.LEVel.AMPLitude
```

Related Objects

SCPI.SOURce.VOLTage.CONTrol.LEVel.STATe

Equivalent Key

PN Menu: **DC Control Voltage** > **DC Control Voltage**

SP Menu: **DC Control Voltage** > **DC Control Voltage**

FP Menu: **DC Control Voltage** > **DC Control Voltage**

TR Menu: **DC Control Voltage** > **DC Control Voltage**

AM Menu: **DC Control Voltage** > **DC Control Voltage**

BB Menu: **DC Control Voltage** > **DC Control Voltage**

USER Menu: **DC Control Voltage** > **DC Control Voltage**

Equivalent SCPI Command

Syntax

:SOURce:VOLTage:CONTrol:LEVel:AMPLitude {-15 ~ 35}

:SOURce:VOLTage:CONTrol:LEVel:AMPLitude?

Query Response

{-15 ~ 35} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:VOLTage:CONTrol:LEVel:AMPLitude 0"
20 OUTPUT 717;":SOURce:VOLTage:CONTrol:LEVel:AMPLitude ?"
30 ENTER 717;A
```

SCPI.SOURce.VOLTage.CONTrol.LEVel.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.VOLTage.CONTrol.LEVel.STATe = *Value*

Value = SCPI.SOURce.VOLTage.CONTrol.LEVel.STATe

Description

This command sets/gets fixed voltage control On/Off at voltage power sweep.

Variable

Parameter	*Value*
Description	Fixed voltage control On/Off at voltage power sweep
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.SOURce.VOLTage.CONTrol.LEVel.STATe = Var
Var = SCPI.SOURce.VOLTage.CONTrol.LEVel.STATe
```

Related Objects

SCPI.SOURce.VOLTage.CONTrol.LEVel.AMPLitude

Equivalent Key

PN Menu: **DC Control Voltage** > **DC Control Output**

SP Menu: **DC Control Voltage** > **DC Control Output**

FP Menu: **DC Control Voltage** > **DC Control Output**

TR Menu: **DC Control Voltage** > **DC Control Output**

AM Menu: **DC Control Voltage** > **DC Control Output**

BB Menu: **DC Control Voltage** > **DC Control Output**

USER Menu: **DC Control Voltage** > **DC Control Output**

Equivalent SCPI Command

Syntax

:SOURce:VOLTage:CONTrol:LEVel:STATe {ON|OFF|1|0}

:SOURce:VOLTage:CONTrol:LEVel:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:VOLTage:CONTrol:LEVel:STATe 1"
20 OUTPUT 717;":SOURce:VOLTage:CONTrol:LEVel:STATe ?"
30 ENTER 717;A
```

SCPI.SOURce.VOLTage.CONTrol.LIMit.HIGH

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.VOLTage.CONTrol.LIMit.HIGH = *Value*

Value = SCPI.SOURce.VOLTage.CONTrol.LIMit.HIGH

Description

This command sets/gets fixed Voltage control high limit, Resolution 0.1mV.

Variable

Parameter	*Value*
Description	Fixed Voltage control high limit, Resolution 0.1mV
Data Type	Double precision floating point type (Double)
Range	-15 ~ 35
Preset Value	35
Unit	V
Resolution	100u
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SOURce.VOLTage.CONTrol.LIMit.HIGH = Var
Var = SCPI.SOURce.VOLTage.CONTrol.LIMit.HIGH
```

Related Objects

SCPI.SOURce.VOLTage.CONTrol.LIMit.LOW

Equivalent Key

PN Menu: **DC Control Voltage** > **Max Ctrl Voltage Limit**

SP Menu: **DC Control Voltage** > **Max Ctrl Voltage Limit**

FP Menu: **DC Control Voltage** > **Max Ctrl Voltage Limit**

TR Menu: **DC Control Voltage** > **Max Ctrl Voltage Limit**

AM Menu: **DC Control Voltage** > **Max Ctrl Voltage Limit**

BB Menu: **DC Control Voltage** > **Max Ctrl Voltage Limit**

USER Menu: **DC Control Voltage** > **Max Ctrl Voltage Limit**

Equivalent SCPI Command

Syntax

:SOURce:VOLTage:CONTrol:LIMit:HIGH {-15 ~ 35}

:SOURce:VOLTage:CONTrol:LIMit:HIGH?

Query Response

{-15 ~ 35} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:VOLTage:CONTrol:LIMit:HIGH 0"
20 OUTPUT 717;":SOURce:VOLTage:CONTrol:LIMit:HIGH ?"
30 ENTER 717;A
```

SCPI.SOURce.VOLTage.CONTrol.LIMit.LOW

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.VOLTage.CONTrol.LIMit.LOW = *Value*

Value = SCPI.SOURce.VOLTage.CONTrol.LIMit.LOW

Description

This command sets/gets fixed voltage control low limit, Resolution 0.1mdV.

Variable

Parameter	*Value*
Description	Fixed Voltage control low limit, Resolution 0.1mdV
Data Type	Double precision floating point type (Double)
Range	-15 ~ 35
Preset Value	-15
Unit	V
Resolution	100u
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SOURce.VOLTage.CONTrol.LIMit.LOW = Var
Var = SCPI.SOURce.VOLTage.CONTrol.LIMit.LOW
```

Related Objects

SCPI.SOURce.VOLTage.CONTrol.LIMit.HIGH

Equivalent Key

PN Menu: **DC Control Voltage** > **Min Ctrl Voltage Limit**

SP Menu: **DC Control Voltage** > **Min Ctrl Voltage Limit**

FP Menu: **DC Control Voltage** > **Min Ctrl Voltage Limit**

TR Menu: **DC Control Voltage** > **Min Ctrl Voltage Limit**

AM Menu: **DC Control Voltage** > **Min Ctrl Voltage Limit**

BB Menu: **DC Control Voltage** > **Min Ctrl Voltage Limit**

USER Menu: **DC Control Voltage** > **Min Ctrl Voltage Limit**

Equivalent SCPI Command

Syntax

:SOURce:VOLTage:CONTrol:LIMit:LOW {-15 ~ 35}

:SOURce:VOLTage:CONTrol:LIMit:LOW?

Query Response

{-15 ~ 35} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:VOLTage:CONTrol:LIMit:LOW 0"
20 OUTPUT 717;":SOURce:VOLTage:CONTrol:LIMit:LOW ?"
30 ENTER 717;A
```

SCPI.SOURce.VOLTage.POWer.DELay

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.VOLTage.POWer.DELay = *Value*

Value = SCPI.SOURce.VOLTage.POWer.DELay

Description

This command sets/gets source power setting delay (in sec).

Variable

Parameter	*Value*
Description	Source Power setting delay (in sec)
Data Type	Double precision floating point type (Double)
Range	0 ~ 1
Preset Value	100m
Unit	s
Resolution	1m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SOURce.VOLTage.POWer.DELay = Var
Var = SCPI.SOURce.VOLTage.POWer.DELay
```

Equivalent Key

PN Menu: **DC Power Voltage** > **DC Power Delay**

SP Menu: **DC Power Voltage** > **DC Power Delay**

FP Menu: **DC Power Voltage** > **DC Power Delay**

TR Menu: **DC Power Voltage** > **DC Power Delay**

AM Menu: **DC Power Voltage** > **DC Power Delay**

BB Menu: **DC Power Voltage** > **DC Power Delay**

USER Menu: **DC Power Voltage** > **DC Power Delay**

Equivalent SCPI Command

Syntax

:SOURce:VOLTage:POWer:DELay {0 ~ 1}

:SOURce:VOLTage:POWer:DELay?

Query Response

{0 ~ 1} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:VOLTage:POWer:DELay 0"
20 OUTPUT 717;":SOURce:VOLTage:POWer:DELay ?"
30 ENTER 717;A
```

SCPI.SOURce.VOLTage.POWer.LEVel.AMPLitude

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.VOLTage.POWer.LEVel.AMPLitude = *Value*

Value = SCPI.SOURce.VOLTage.POWer.LEVel.AMPLitude

Description

This command sets/gets fixed voltage power value at voltage control sweep.

Variable

Parameter	*Value*
Description	Fixed voltage power value at voltage control sweep
Data Type	Double precision floating point type (Double)
Range	0 ~ 16
Preset Value	0
Unit	V
Resolution	1m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SOURce.VOLTage.POWer.LEVel.AMPLitude = Var
Var = SCPI.SOURce.VOLTage.POWer.LEVel.AMPLitude
```

Related Objects

SCPI.SOURce.VOLTage.POWer.LEVel.STATe

Equivalent Key

PN Menu: **DC Power Voltage** > **DC Power Voltage**

SP Menu: **DC Power Voltage** > **DC Power Voltage**

FP Menu: **DC Power Voltage** > **DC Power Voltage**

TR Menu: **DC Power Voltage** > **DC Power Voltage**

AM Menu: **DC Power Voltage** > **DC Power Voltage**

BB Menu: **DC Power Voltage** > **DC Power Voltage**

USER Menu: **DC Power Voltage** > **DC Power Voltage**

Equivalent SCPI Command

Syntax

:SOURce:VOLTage:POWer:LEVel:AMPLitude {0 ~ 16}

:SOURce:VOLTage:POWer:LEVel:AMPLitude?

Query Response

{0 ~ 16} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:VOLTage:POWer:LEVel:AMPLitude 0"
20 OUTPUT 717;":SOURce:VOLTage:POWer:LEVel:AMPLitude ?"
30 ENTER 717;A
```

SCPI.SOURce.VOLTage.POWer.LEVel.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.VOLTage.POWer.LEVel.STATe = *Value*

Value = SCPI.SOURce.VOLTage.POWer.LEVel.STATe

Description

This command sets/gets fixed voltage power On/Off at voltage control sweep.

Variable

Parameter	*Value*
Description	fixed voltage power On/Off at voltage control sweep
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.SOURce.VOLTage.POWer.LEVel.STATe = Var
Var = SCPI.SOURce.VOLTage.POWer.LEVel.STATe
```

Related Objects

SCPI.SOURce.VOLTage.POWer.LEVel.AMPLitude

Equivalent Key

PN Menu: **DC Power Voltage** > **DC Power Output**

SP Menu: **DC Power Voltage** > **DC Power Output**

FP Menu: **DC Power Voltage** > **DC Power Output**

TR Menu: **DC Power Voltage** > **DC Power Output**

AM Menu: **DC Power Voltage** > **DC Power Output**

BB Menu: **DC Power Voltage** > **DC Power Output**

USER Menu: **DC Power Voltage** > **DC Power Output**

Equivalent SCPI Command

Syntax

:SOURce:VOLTage:POWer:LEVel:STATe {ON|OFF|1|0}

:SOURce:VOLTage:POWer:LEVel:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:VOLTage:POWer:LEVel:STATe 1"
20 OUTPUT 717;":SOURce:VOLTage:POWer:LEVel:STATe ?"
30 ENTER 717;A
```

SCPI.SOURce.VOLTage.POWer.LIMit.HIGH

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.VOLTage.POWer.LIMit.HIGH = *Value*

Value = SCPI.SOURce.VOLTage.POWer.LIMit.HIGH

Description

This command sets/gets fixed voltage power high limit, Resolution 1mV.

Variable

Parameter	*Value*
Description	Fixed voltage power high limit, Resolution 1mV
Data Type	Double precision floating point type (Double)
Range	0 ~ 16
Preset Value	16
Unit	V
Resolution	1m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SOURce.VOLTage.POWer.LIMit.HIGH = Var
Var = SCPI.SOURce.VOLTage.POWer.LIMit.HIGH
```

Related Objects

SCPI.SOURce.VOLTage.POWer.LIMit.LOW

Equivalent Key

PN Menu: **DC Power Voltage** > **Max Pwr Voltage Limit**

SP Menu: **DC Power Voltage** > **Max Pwr Voltage Limit**

FP Menu: **DC Power Voltage** > **Max Pwr Voltage Limit**

TR Menu: **DC Power Voltage** > **Max Pwr Voltage Limit**

AM Menu: **DC Power Voltage** > **Max Pwr Voltage Limit**

BB Menu: **DC Power Voltage** > **Max Pwr Voltage Limit**

USER Menu: **DC Power Voltage** > **Max Pwr Voltage Limit**

Equivalent SCPI Command

Syntax

:SOURce:VOLTage:POWer:LIMit:HIGH {0 ~ 16}

:SOURce:VOLTage:POWer:LIMit:HIGH?

Query Response

{0 ~ 16} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:VOLTage:POWer:LIMit:HIGH 0"
20 OUTPUT 717;":SOURce:VOLTage:POWer:LIMit:HIGH ?"
30 ENTER 717;A
```

SCPI.SOURce.VOLTage.POWer.LIMit.LOW

Object Type

Property (**Read-Write**)

Syntax

SCPI.SOURce.VOLTage.POWer.LIMit.LOW = *Value*

Value = SCPI.SOURce.VOLTage.POWer.LIMit.LOW

Description

This command sets/gets fixed voltage power low limit, Resolution 1mdV.

Variable

Parameter	*Value*
Description	Fixed voltage power low limit
Data Type	Double precision floating point type (Double)
Range	0 ~ 16
Preset Value	0
Unit	V
Resolution	1m
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.SOURce.VOLTage.POWer.LIMit.LOW = Var
Var = SCPI.SOURce.VOLTage.POWer.LIMit.LOW
```

Related Objects

SCPI.SOURce.VOLTage.POWer.LIMit.HIGH

Equivalent Key

PN Menu: **DC Power Voltage** > **Min Pwr Voltage Limit**

SP Menu: **DC Power Voltage** > **Min Pwr Voltage Limit**

FP Menu: **DC Power Voltage** > **Min Pwr Voltage Limit**

TR Menu: **DC Power Voltage** > **Min Pwr Voltage Limit**

AM Menu: **DC Power Voltage** > **Min Pwr Voltage Limit**

BB Menu: **DC Power Voltage** > **Min Pwr Voltage Limit**

USER Menu: **DC Power Voltage** > **Min Pwr Voltage Limit**

Equivalent SCPI Command

Syntax

:SOURce:VOLTage:POWer:LIMit:LOW {0 ~ 16}

:SOURce:VOLTage:POWer:LIMit:LOW?

Query Response

{0 ~ 16} <newline><^END>

Example of use

```
10 OUTPUT 717;":SOURce:VOLTage:POWer:LIMit:LOW 0"
20 OUTPUT 717;":SOURce:VOLTage:POWer:LIMit:LOW ?"
30 ENTER 717;A
```

Status

SCPI.STATus.OPERation.BIT12.CLEar

Object Type

Method (**Write Only**)

Syntax

SCPI.STATus.OPERation.BIT12.CLEar = *Value*

Description

This command clears specified bit of BIT12 condition register.

NOTE Parameter is bit position number.

Variable

Parameter	*Value*
Description	Clears specified bit of BIT12 condition register
Data Type	Long integer type (Long)
Range	0 ~ 14
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.OPERation.BIT12.CLEar = Var
```

Related Objects

SCPI.STATus.OPERation.BIT12.CONDition

SCPI.STATus.OPERation.BIT12.ENABle

SCPI.STATus.OPERation.BIT12.EVENt

SCPI.STATus.OPERation.BIT12.NTRansition

SCPI.STATus.OPERation.BIT12.PTRansition

SCPI.STATus.OPERation.BIT12.SET

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:OPERation:BIT12:CLEar {0 ~ 14}

Query Response

{0 ~ 14} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:OPERation:BIT12:CLEar 0 "
20 ENTER 717;A
```

SCPI.STATus.OPERation.BIT12.CONDition

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.OPERation.BIT12.CONDition

Description

This command gets operation-program status register.

NOTE Effective only for 0-14 bits..

Variable

Parameter	*Value*
Description	Operation-program status register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.OPERation.BIT12.CONDition
```

Related Objects

SCPI.STATus.OPERation.BIT12.CLEar

SCPI.STATus.OPERation.BIT12.ENABle

SCPI.STATus.OPERation.BIT12.EVENt

SCPI.STATus.OPERation.BIT12.NTRansition

SCPI.STATus.OPERation.BIT12.PTRansition

SCPI.STATus.OPERation.BIT12.SET

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:OPERation:BIT12:CONDition?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:OPERation:BIT12:CONDition ?"
20 ENTER 717;A
```

SCPI.STATus.OPERation.BIT12.ENABle

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.OPERation.BIT12.ENABle = *Value*

Value = SCPI.STATus.OPERation.BIT12.ENABle

Description

This command sets/gets operation-program status enable register.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Operation-program status enable register
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.OPERation.BIT12.ENABle = Var
Var = SCPI.STATus.OPERation.BIT12.ENABle
```

Related Objects

SCPI.STATus.OPERation.BIT12.CLEar

SCPI.STATus.OPERation.BIT12.CONDition

SCPI.STATus.OPERation.BIT12.EVENt

SCPI.STATus.OPERation.BIT12.NTRansition

SCPI.STATus.OPERation.BIT12.PTRansition

SCPI.STATus.OPERation.BIT12.SET

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:OPERation:BIT12:ENABle {0 ~ 65535}

:STATus:OPERation:BIT12:ENABle?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:OPERation:BIT12:ENABle 0 "
20 OUTPUT 717;":STATus:OPERation:BIT12:ENABle ?"
30 ENTER 717;A
```

SCPI.STATus.OPERation.BIT12.EVENt

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.OPERation.BIT12.EVENt

Description

This command gets operation-program status event register.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Operation-program status event register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.OPERation.BIT12.EVENt
```

Related Objects

SCPI.STATus.OPERation.BIT12.CLEar

SCPI.STATus.OPERation.BIT12.CONDition

SCPI.STATus.OPERation.BIT12.ENABle

SCPI.STATus.OPERation.BIT12.NTRansition

SCPI.STATus.OPERation.BIT12.PTRansition

SCPI.STATus.OPERation.BIT12.SET

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:OPERation:BIT12[:EVENt]?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:OPERation:BIT12[:EVENt] ?"
20 ENTER 717;A
```

SCPI.STATus.OPERation.BIT12.NTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.OPERation.BIT12.NTRansition = *Value*

Value = SCPI.STATus.OPERation.BIT12.NTRansition

Description

This command sets/gets operation-program status negative transition filter.

NOTE Effective only for 0-14 bits..

Variable

Parameter	*Value*
Description	Operation-program status negative transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.OPERation.BIT12.NTRansition = Var
Var = SCPI.STATus.OPERation.BIT12.NTRansition
```

Related Objects

SCPI.STATus.OPERation.BIT12.CLEar

SCPI.STATus.OPERation.BIT12.CONDition

SCPI.STATus.OPERation.BIT12.ENABle

SCPI.STATus.OPERation.BIT12.EVENt

SCPI.STATus.OPERation.BIT12.PTRansition

SCPI.STATus.OPERation.BIT12.SET

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:OPERation:BIT12:NTRansition {0 ~ 65535}

:STATus:OPERation:BIT12:NTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:OPERation:BIT12:NTRansition 0 "
20 OUTPUT 717;":STATus:OPERation:BIT12:NTRansition ?"
30 ENTER 717;A
```

SCPI.STATus.OPERation.BIT12.PTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.OPERation.BIT12.PTRansition = *Value*

Value = SCPI.STATus.OPERation.BIT12.PTRansition

Description

This command sets/gets operation-program status positive transition filter.

NOTE Effective only for 0-14 bits. (*1).

Variable

Parameter	*Value*
Description	Operation-program status positive transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	32767
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.OPERation.BIT12.PTRansition = Var
Var = SCPI.STATus.OPERation.BIT12.PTRansition
```

Related Objects

SCPI.STATus.OPERation.BIT12.CLEar

SCPI.STATus.OPERation.BIT12.CONDition

SCPI.STATus.OPERation.BIT12.ENABle

SCPI.STATus.OPERation.BIT12.EVENt

SCPI.STATus.OPERation.BIT12.NTRansition

SCPI.STATus.OPERation.BIT12.SET

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:OPERation:BIT12:PTRansition {0 ~ 65535}

:STATus:OPERation:BIT12:PTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:OPERation:BIT12:PTRansition 0 "
20 OUTPUT 717;":STATus:OPERation:BIT12:PTRansition ?"
30 ENTER 717;A
```

SCPI.STATus.OPERation.BIT12.SET

Object Type

Method (**Write Only**)

Syntax

SCPI.STATus.OPERation.BIT12.SET = *Value*

Description

This command sets specified bit of BIT12 condition register.

NOTE Parameter is bit position number.

Variable

Parameter	*Value*
Description	Specified bit of BIT12 condition register
Data Type	Long integer type (Long)
Range	0 ~ 14
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.OPERation.BIT12.SET = Var
```

Related Objects

SCPI.STATus.OPERation.BIT12.CLEar

SCPI.STATus.OPERation.BIT12.CONDition

SCPI.STATus.OPERation.BIT12.ENABle

SCPI.STATus.OPERation.BIT12.EVENt

SCPI.STATus.OPERation.BIT12.NTRansition

SCPI.STATus.OPERation.BIT12.PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:OPERation:BIT12:SET {0 ~ 14}

Query Response

{0 ~ 14} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:OPERation:BIT12:SET 0 "
20 ENTER 717;A
```

SCPI.STATus.OPERation.CONDition

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.OPERation.CONDition

Description

This command gets operation status register.

NOTE Effective only for 0-14 bits..

Variable

Parameter	*Value*
Description	operation status register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.OPERation.CONDition
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:OPERation:CONDition?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:OPERation:CONDition ?"
20 ENTER 717;A
```

SCPI.STATus.OPERation.ENABle

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.OPERation.ENABle = *Value*

Value = SCPI.STATus.OPERation.ENABle

Description

This command sets/gets operation status enable register.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Operation status enable register
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.OPERation.ENABle = Var
Var = SCPI.STATus.OPERation.ENABle
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:OPERation:ENABle {0 ~ 65535}

:STATus:OPERation:ENABle?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:OPERation:ENABle 0 "
20 OUTPUT 717;":STATus:OPERation:ENABle ?"
30 ENTER 717;A
```

SCPI.STATus.OPERation.EVENt

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.OPERation.EVENt

Description

This command gets operation status event register.

NOTE Effective only for 0-14 bits..

Variable

Parameter	*Value*
Description	Operation status event register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.OPERation.EVENt
```

Related Objects

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:OPERation[:EVENt]?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:OPERation[:EVENt] ?"
20 ENTER 717;A
```

SCPI.STATus.OPERation.NTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.OPERation.NTRansition = *Value*

Value = SCPI.STATus.OPERation.NTRansition

Description

This command sets/gets operation status negative transition filter.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Operation status negative transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.OPERation.NTRansition = Var
Var = SCPI.STATus.OPERation.NTRansition
```

Related Objects

SCPI.STATus.OPERation.PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:OPERation:NTRansition {0 ~ 65535}

:STATus:OPERation:NTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:OPERation:NTRansition 0 "
20 OUTPUT 717;":STATus:OPERation:NTRansition ?"
30 ENTER 717;A
```

SCPI.STATus.OPERation.PTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.OPERation.PTRansition = *Value*

Value = SCPI.STATus.OPERation.PTRansition

Description

This command sets/gets operation status positive transition filter.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Operation status positive transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	32767
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.OPERation.PTRansition = Var
Var = SCPI.STATus.OPERation.PTRansition
```

Related Objects

SCPI.STATus.OPERation.NTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:OPERation:PTRansition {0 ~ 65535}

:STATus:OPERation:PTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:OPERation:PTRansition 0 "
20 OUTPUT 717;":STATus:OPERation:PTRansition ?"
30 ENTER 717;A
```

SCPI.STATus.PRESet

Object Type

Method (**Write Only**)

Syntax

SCPI.STATus.PRESet

Description

This command presets status registers.

Examples

SCPI.STATus.PRESet

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:PRESet

Example of use

```
10 OUTPUT 717;":STATus:PRESet"
```

SCPI.STATus.QUEStionable.CONDition

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.CONDition

Description

This command gets questionable status register.

NOTE Effective only for 0-14 bits..

Variable

Parameter	*Value*
Description	Questionable status register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.CONDition
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:CONDition?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:CONDition ?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.CURRent.ENABle

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.CURRent.ENABle = *Value*

Value = SCPI.STATus.QUEStionable.CURRent.ENABle

Description

This command sets/gets questionable-current status enable register.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-current status enable register
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.CURRent.ENABle = Var
Var = SCPI.STATus.QUEStionable.CURRent.ENABle
```

Related Objects

SCPI.STATus.QUEStionable.CURRent.EVENt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:CURRent:ENABle {0 ~ 65535}

:STATus:QUEStionable:CURRent:ENABle?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:CURRent:ENABle 0 "
20 OUTPUT 717;":STATus:QUEStionable:CURRent:ENABle ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.CURRent.EVENt

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.CURRent.EVENt

Description

This command gets questionable-current status event register.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-current status event register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.CURRent.EVENt
```

Related Objects

SCPI.STATus.QUEStionable.CURRent.ENABle

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:CURRent[:EVENt]?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:CURRent[:EVENt] ?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.DCONverter.ENABle

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.DCONverter.ENABle = *Value*

Value = SCPI.STATus.QUEStionable.DCONverter.ENABle

Description

This command sets/gets questionable-downconverter status enable register.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-downconverter status enable register
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.DCONverter.ENABle = Var
Var = SCPI.STATus.QUEStionable.DCONverter.ENABle
```

Related Objects

SCPI.STATus.QUEStionable.DCONverter.EVENt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:DCONverter:ENABle {0 ~ 65535}

:STATus:QUEStionable:DCONverter:ENABle?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:DCONverter:ENABle 0 "
20 OUTPUT 717;":STATus:QUEStionable:DCONverter:ENABle ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.DCONverter.EVENt

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.DCONverter.EVENt

Description

This command gets questionable-downconverter status event register.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-downconverter status event register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.DCONverter.EVENt
```

Related Objects

SCPI.STATus.QUEStionable.DCONverter.ENABle

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:DCONverter[:EVENt]?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:DCONverter[:EVENt] ?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.ENABle

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.ENABle = *Value*

Value = SCPI.STATus.QUEStionable.ENABle

Description

This command sets/gets questionable status enable register.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable status enable register
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.ENABle = Var
Var = SCPI.STATus.QUEStionable.ENABle
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:ENABle {0 ~ 65535}

:STATus:QUEStionable:ENABle?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:ENABle 0 "
20 OUTPUT 717;":STATus:QUEStionable:ENABle ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.EVENt

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.EVENt

Description

This command gets questionable status event register.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable status event register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.EVENt
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable[:EVENt]?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable[:EVENt] ?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.AM(Ch).CONDition

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.LIMit.AM(Ch).CONDition

Description

This command gets questionable-limit-am1 status register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-am1 status register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.LIMit.AM1.CONDition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.AM(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.AM(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.AM(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.AM(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:AM[1-1]:CONDition?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:AM1:CONDition ?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.AM(Ch).ENABle

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.AM(Ch).ENABle = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.AM(Ch).ENABle

Description

This command sets/gets questionable-limit-am1 status enable register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-am1 status enable register
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.AM1.ENABle = Var
Var = SCPI.STATus.QUEStionable.LIMit.AM1.ENABle
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.AM(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.AM(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.AM(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.AM(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:AM[1-1]:ENABle {0 ~ 65535}

:STATus:QUEStionable:LIMit:AM[1-1]:ENABle?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:AM1:ENABle 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:AM1:ENABle ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.AM(Ch).EVENt

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.LIMit.AM(Ch).EVENt

Description

This command gets questionable-limit-am1 status event register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-am1 status event register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.LIMit.AM1.EVENt
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.AM(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.AM(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.AM(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.AM(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:AM[1-1][:EVENt]?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:AM1[:EVENt] ?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.AM(Ch).NTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.AM(Ch).NTRansition = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.AM(Ch).NTRansition

Description

This command sets/gets questionable-limit-am1 status negative transition filter, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-am1 status negative transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.AM1.NTRansition = Var
Var = SCPI.STATus.QUEStionable.LIMit.AM1.NTRansition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.AM(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.AM(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.AM(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.AM(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:AM[1-1]:NTRansition {0 ~ 65535}

:STATus:QUEStionable:LIMit:AM[1-1]:NTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:AM1:NTRansition 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:AM1:NTRansition ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.AM(Ch).PTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.AM(Ch).PTRansition = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.AM(Ch).PTRansition

Description

This command sets/gets questionable-limit-am1 status positive transition filter, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-am1 status positive transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	32767
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.AM1.PTRansition = Var
Var = SCPI.STATus.QUEStionable.LIMit.AM1.PTRansition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.AM(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.AM(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.AM(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.AM(Ch).NTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:AM[1-1]:PTRansition {0 ~ 65535}

:STATus:QUEStionable:LIMit:AM[1-1]:PTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:AM1:PTRansition 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:AM1:PTRansition ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.BB(Ch).CONDition

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.LIMit.BB(Ch).CONDition

Description

This command gets questionable-limit-bb1 status register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-bb1 status register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.LIMit.BB1.CONDition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.BB(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.BB(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.BB(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.BB(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:BB[1-1]:CONDition?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:BB1:CONDition ?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.BB(Ch).ENABle

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.BB(Ch).ENABle = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.BB(Ch).ENABle

Description

This command sets/gets questionable-limit-bb1 status enable register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-bb1 status enable register
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.BB1.ENABle = Var
Var = SCPI.STATus.QUEStionable.LIMit.BB1.ENABle
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.BB(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.BB(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.BB(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.BB(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:BB[1-1]:ENABle {0 ~ 65535}

:STATus:QUEStionable:LIMit:BB[1-1]:ENABle?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:BB1:ENABle 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:BB1:ENABle ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.BB(Ch).EVENt

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.LIMit.BB(Ch).EVENt

Description

This command gets questionable-limit-bb1 status event register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-bb1 status event register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.LIMit.BB1.EVENt
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.BB(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.BB(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.BB(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.BB(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:BB[1-1][:EVENt]?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:BB1[:EVENt] ?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.BB(Ch).NTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.BB(Ch).NTRansition = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.BB(Ch).NTRansition

Description

This command sets/gets questionable-limit-bb1 status negative transition filter, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-bb1 status negative transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.BB1.NTRansition = Var
Var = SCPI.STATus.QUEStionable.LIMit.BB1.NTRansition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.BB(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.BB(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.BB(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.BB(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:BB[1-1]:NTRansition {0 ~ 65535}

:STATus:QUEStionable:LIMit:BB[1-1]:NTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:BB1:NTRansition 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:BB1:NTRansition ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.BB(Ch).PTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.BB(Ch).PTRansition = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.BB(Ch).PTRansition

Description

This command sets/gets questionable-limit-bb1 status positive transition filter, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-bb1 status positive transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	32767
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.BB1.PTRansition = Var
Var = SCPI.STATus.QUEStionable.LIMit.BB1.PTRansition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.BB(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.BB(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.BB(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.BB(Ch).NTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:BB[1-1]:PTRansition {0 ~ 65535}

:STATus:QUEStionable:LIMit:BB[1-1]:PTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:BB1:PTRansition 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:BB1:PTRansition ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.CONDition

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.LIMit.CONDition

Description

This command gets questionable-limit status register.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit status register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.LIMit.CONDition
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:CONDition?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:CONDition ?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.ENABle

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.ENABle = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.ENABle

Description

This command sets/gets questionable-limit status enable register.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit status enable register
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.ENABle = Var
Var = SCPI.STATus.QUEStionable.LIMit.ENABle
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:ENABle {0 ~ 65535}

:STATus:QUEStionable:LIMit:ENABle?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:ENABle 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:ENABle ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.EVENt

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.LIMit.EVENt

Description

This command gets questionable-limit status event register.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit status event register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.LIMit.EVENt
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit[:EVENt]?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit[:EVENt] ?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.FP(Ch).CONDition

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.LIMit.FP(Ch).CONDition

Description

This command gets questionable-limit-fp1 status register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-fp1 status register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.LIMit.FP1.CONDition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.FP(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.FP(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.FP(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.FP(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:FP[1-1]:CONDition?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:FP1:CONDition ?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.FP(Ch).ENABle

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.FP(Ch).ENABle = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.FP(Ch).ENABle

Description

This command sets/gets questionable-limit-fp1 status enable register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-fp1 status enable register
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.FP1.ENABle = Var
Var = SCPI.STATus.QUEStionable.LIMit.FP1.ENABle
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.FP(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.FP(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.FP(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.FP(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:FP[1-1]:ENABle {0 ~ 65535}

:STATus:QUEStionable:LIMit:FP[1-1]:ENABle?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:FP1:ENABle 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:FP1:ENABle ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.FP(Ch).EVENt

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.LIMit.FP(Ch).EVENt

Description

This command gets questionable-limit-fp1 status event register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-fp1 status event register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.LIMit.FP1.EVENt
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.FP(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.FP(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.FP(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.FP(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:FP[1-1][:EVENt]?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:FP1[:EVENt] ?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.FP(Ch).NTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.FP(Ch).NTRansition = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.FP(Ch).NTRansition

Description

This command sets/gets questionable-limit-fp1 status negative transition filter, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-fp1 status negative transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.FP1.NTRansition = Var
Var = SCPI.STATus.QUEStionable.LIMit.FP1.NTRansition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.FP(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.FP(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.FP(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.FP(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:FP[1-1]:NTRansition {0 ~ 65535}

:STATus:QUEStionable:LIMit:FP[1-1]:NTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:FP1:NTRansition 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:FP1:NTRansition ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.FP(Ch).PTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.FP(Ch).PTRansition = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.FP(Ch).PTRansition

Description

This command sets/gets questionable-limit-fp1 status positive transition filter, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-fp1 status positive transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	32767
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.FP1.PTRansition = Var
Var = SCPI.STATus.QUEStionable.LIMit.FP1.PTRansition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.FP(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.FP(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.FP(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.FP(Ch).NTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:FP[1-1]:PTRansition {0 ~ 65535}

:STATus:QUEStionable:LIMit:FP[1-1]:PTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:FP1:PTRansition 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:FP1:PTRansition ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.NTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.NTRansition = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.NTRansition

Description

This command sets/gets questionable-limit status negative transition filter.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit status negative transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.NTRansition = Var
Var = SCPI.STATus.QUEStionable.LIMit.NTRansition
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:NTRansition {0 ~ 65535}

:STATus:QUEStionable:LIMit:NTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:NTRansition 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:NTRansition ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.PN(Ch).CONDition

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.LIMit.PN(Ch).CONDition

Description

This command gets questionable-limit-pn1 status register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-pn1 status register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.LIMit.PN1.CONDition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.PN(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.PN(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.PN(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.PN(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:PN[1-1]:CONDition?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:PN1:CONDition ?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.PN(Ch).ENABle

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.PN(Ch).ENABle = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.PN(Ch).ENABle

Description

This command sets/gets questionable-limit-pn1 status enable register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-pn1 status enable register
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.PN1.ENABle = Var
Var = SCPI.STATus.QUEStionable.LIMit.PN1.ENABle
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.PN(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.PN(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.PN(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.PN(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:PN[1-1]:ENABle {0 ~ 65535}

:STATus:QUEStionable:LIMit:PN[1-1]:ENABle?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:PN1:ENABle 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:PN1:ENABle?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.PN(Ch).EVENt

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.LIMit.PN(Ch).EVENt

Description

This command gets questionable-limit-pn1 status event register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-pn1 status event register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.LIMit.PN1.EVENt
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.PN(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.PN(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.PN(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.PN(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:PN[1-1][:EVENt]?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:PN1[:EVENt] ?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.PN(Ch).NTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.PN(Ch).NTRansition = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.PN(Ch).NTRansition

Description

This command sets/gets questionable-limit-pn1 status negative transition filter, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-pn1 status negative transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.PN1.NTRansition = Var
Var = SCPI.STATus.QUEStionable.LIMit.PN1.NTRansition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.PN(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.PN(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.PN(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.PN(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:PN[1-1]:NTRansition {0 ~ 65535}

:STATus:QUEStionable:LIMit:PN[1-1]:NTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:PN1:NTRansition 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:PN1:NTRansition ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.PN(Ch).PTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.PN(Ch).PTRansition = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.PN(Ch).PTRansition

Description

This command sets/gets questionable-limit-pn1 status positive transition filter, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-pn1 status positive transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	32767
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.PN1.PTRansition = Var
Var = SCPI.STATus.QUEStionable.LIMit.PN1.PTRansition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.PN(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.PN(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.PN(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.PN(Ch).NTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:PN[1-1]:PTRansition {0 ~ 65535}

:STATus:QUEStionable:LIMit:PN[1-1]:PTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:PN1:PTRansition 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:PN1:PTRansition ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.PS(Ch).CONDition

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.LIMit.PS(Ch).CONDition

Description

This command gets questionable-limit-pn1 status register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-pn1 status register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.LIMit.PS1.CONDition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.PN(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.PN(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.PN(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.PN(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:PS[1-1]:CONDition?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:PS1:CONDition?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.PS(Ch).ENABle

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.PS(Ch).ENABle = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.PS(Ch).ENABle

Description

This command sets/gets questionable-limit-pn1 status enable register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-pn1 status enable register
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.PS1.ENABle = Var
Var = SCPI.STATus.QUEStionable.LIMit.PS1.ENABle
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.PS(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.PS(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.PS(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.PS(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:PS[1-1]:ENABle {0 ~ 65535}

:STATus:QUEStionable:LIMit:PS[1-1]:ENABle?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:PS1:ENABle 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:PS1:ENABle?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.PS(Ch).EVENt

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.LIMit.PS(Ch).EVENt

Description

This command gets questionable-limit-pn1 status event register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-pn1 status event register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var = SCPI.STATus.QUEStionable.LIMit.PS1.EVENt
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.PS(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.PS(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.PS(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.PS(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:PS[1-1][:EVENt]?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:PS1[:EVENt]?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.PS(Ch).NTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.PS(Ch).NTRansition = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.PS(Ch).NTRansition

Description

This command sets/gets questionable-limit-pn1 status negative transition filter, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-pn1 status negative transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.PS1.NTRansition = Var
Var = SCPI.STATus.QUEStionable.LIMit.PS1.NTRansition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.PS(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.PS(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.PS(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.PS(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:PS[1-1]:NTRansition {0 ~ 65535}

:STATus:QUEStionable:LIMit:PS[1-1]:NTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:PS1:NTRansition 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:PS1:NTRansition?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.PS(Ch).PTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.PS(Ch).PTRansition = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.PS(Ch).PTRansition

Description

This command sets/gets questionable-limit-pn1 status positive transition filter, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-pn1 status positive transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	32767
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.PS1.PTRansition = Var
Var = SCPI.STATus.QUEStionable.LIMit.PS1.PTRansition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.PS(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.PS(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.PS(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.PS(Ch).NTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:PS[1-1]:PTRansition {0 ~ 65535}

:STATus:QUEStionable:LIMit:PS[1-1]:PTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:PS1:PTRansition 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:PS1:PTRansition?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.PTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.PTRansition = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.PTRansition

Description

This command sets/gets questionable-limit status positive transition filter.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit status positive transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	32767
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.PTRansition = Var
Var = SCPI.STATus.QUEStionable.LIMit.PTRansition
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:PTRansition {0 ~ 65535}

:STATus:QUEStionable:LIMit:PTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:PTRansition 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:PTRansition ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.SP(Ch).CONDition

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.LIMit.SP(Ch).CONDition

Description

This command gets questionable-limit-sp1 status register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-sp1 status register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.LIMit.SP1.CONDition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.SP(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.SP(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.SP(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.SP(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:SP[1-1]:CONDition?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:SP1:CONDition ?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.SP(Ch).ENABle

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.SP(Ch).ENABle = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.SP(Ch).ENABle

Description

This command sets/gets questionable-limit-sp1 status enable register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-sp1 status enable register
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.SP1.ENABle = Var
Var = SCPI.STATus.QUEStionable.LIMit.SP1.ENABle
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.SP(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.SP(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.SP(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.SP(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:SP[1-1]:ENABle {0 ~ 65535}

:STATus:QUEStionable:LIMit:SP[1-1]:ENABle?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:SP1:ENABle 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:SP1:ENABle ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.SP(Ch).EVENt

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.LIMit.SP(Ch).EVENt

Description

This command gets questionable-limit-sp1 status event register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-sp1 status event register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.LIMit.SP1.EVENt
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.SP(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.SP(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.SP(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.SP(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:SP[1-1][:EVENt]?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:SP1[:EVENt] ?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.SP(Ch).NTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.SP(Ch).NTRansition = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.SP(Ch).NTRansition

Description

This command sets/gets questionable-limit-sp1 status negative transition filter, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-sp1 status negative transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.SP1.NTRansition = Var
Var = SCPI.STATus.QUEStionable.LIMit.SP1.NTRansition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.SP(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.SP(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.SP(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.SP(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:SP[1-1]:NTRansition {0 ~ 65535}

:STATus:QUEStionable:LIMit:SP[1-1]:NTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:SP1:NTRansition 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:SP1:NTRansition ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.SP(Ch).PTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.SP(Ch).PTRansition = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.SP(Ch).PTRansition

Description

This command sets/gets questionable-limit-sp1 status positive transition filter, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-sp1 status positive transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	32767
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.SP1.PTRansition = Var
Var = SCPI.STATus.QUEStionable.LIMit.SP1.PTRansition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.SP(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.SP(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.SP(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.SP(Ch).NTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:SP[1-1]:PTRansition {0 ~ 65535}

:STATus:QUEStionable:LIMit:SP[1-1]:PTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:SP1:PTRansition 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:SP1:PTRansition ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.TR(Ch).CONDition

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.LIMit.TR(Ch).CONDition

Description

This command gets questionable-limit-tr1 status register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-tr1 status register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.LIMit.TR1.CONDition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.TR(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.TR(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.TR(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.TR(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:TR[1-1]:CONDition?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:TR1:CONDition ?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.TR(Ch).ENABle

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.TR(Ch).ENABle = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.TR(Ch).ENABle

Description

This command sets/gets questionable-limit-tr1 status enable register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-tr1 status enable register
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.TR1.ENABle = Var
Var = SCPI.STATus.QUEStionable.LIMit.TR1.ENABle
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.TR(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.TR(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.TR(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.TR(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:TR[1-1]:ENABle {0 ~ 65535}

:STATus:QUEStionable:LIMit:TR[1-1]:ENABle?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:TR1:ENABle 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:TR1:ENABle ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.TR(Ch).EVENt

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.LIMit.TR(Ch).EVENt

Description

This command gets questionable-limit-tr1 status event register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-tr1 status event register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.LIMit.TR1.EVENt
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.TR(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.TR(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.TR(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.TR(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:TR[1-1][:EVENt]?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:TR1[:EVENt] ?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.TR(Ch).NTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.TR(Ch).NTRansition = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.TR(Ch).NTRansition

Description

This command sets/gets questionable-limit-tr1 status negative transition filter, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-tr1 status negative transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.TR1.NTRansition = Var
Var = SCPI.STATus.QUEStionable.LIMit.TR1.NTRansition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.TR(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.TR(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.TR(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.TR(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:TR[1-1]:NTRansition {0 ~ 65535}

:STATus:QUEStionable:LIMit:TR[1-1]:NTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:TR1:NTRansition 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:TR1:NTRansition ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.TR(Ch).PTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.TR(Ch).PTRansition = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.TR(Ch).PTRansition

Description

This command sets/gets questionable-limit-tr1 status positive transition filter, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-tr1 status positive transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	32767
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.TR1.PTRansition = Var
Var = SCPI.STATus.QUEStionable.LIMit.TR1.PTRansition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.TR(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.TR(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.TR(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.TR(Ch).NTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:TR[1-1]:PTRansition {0 ~ 65535}

:STATus:QUEStionable:LIMit:TR[1-1]:PTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:TR1:PTRansition 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:TR1:PTRansition ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.USER(Ch).CONDition

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.LIMit.USER(Ch).CONDition

Description

This command gets questionable-limit-User1 status register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-User1 status register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.LIMit.USER1.CONDition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.USER(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.USER(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.USER(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.USER(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:USER[1-1]:CONDition?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:USER1:CONDition ?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.USER(Ch).ENABle

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.USER(Ch).ENABle = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.USER(Ch).ENABle

Description

This command sets/gets questionable-limit-User1 status enable register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-User1 status enable register
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.USER1.ENABle = Var
Var = SCPI.STATus.QUEStionable.LIMit.USER1.ENABle
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.USER(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.USER(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.USER(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.USER(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:USER[1-1]:ENABle {0 ~ 65535}

:STATus:QUEStionable:LIMit:USER[1-1]:ENABle?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:USER1:ENABle 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:USER1:ENABle ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.USER(Ch).EVENt

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.LIMit.USER(Ch).EVENt

Description

This command gets questionable-limit-User1 status event register, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-User1 status event register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.LIMit.USER1.EVENt
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.USER(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.USER(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.USER(Ch).NTRansition

SCPI.STATus.QUEStionable.LIMit.USER(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:USER[1-1][:EVENt]?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:USER1[:EVENt] ?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.USER(Ch).NTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.USER(Ch).NTRansition = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.USER(Ch).NTRansition

Description

This command sets/gets questionable-limit-User1 status negative transition filter, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-User1 status negative transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.USER1.NTRansition = Var
Var = SCPI.STATus.QUEStionable.LIMit.USER1.NTRansition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.USER(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.USER(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.USER(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.USER(Ch).PTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:USER[1-1]:NTRansition {0 ~ 65535}

:STATus:QUEStionable:LIMit:USER[1-1]:NTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:USER1:NTRansition 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:USER1:NTRansition ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.LIMit.USER(Ch).PTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.LIMit.USER(Ch).PTRansition = *Value*

Value = SCPI.STATus.QUEStionable.LIMit.USER(Ch).PTRansition

Description

This command sets/gets questionable-limit-User1 status positive transition filter, for the selected channel *Ch*.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-limit-User1 status positive transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	32767
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.LIMit.USER1.PTRansition = Var
Var = SCPI.STATus.QUEStionable.LIMit.USER1.PTRansition
```

Related Objects

SCPI.STATus.QUEStionable.LIMit.USER(Ch).CONDition

SCPI.STATus.QUEStionable.LIMit.USER(Ch).ENABle

SCPI.STATus.QUEStionable.LIMit.USER(Ch).EVENt

SCPI.STATus.QUEStionable.LIMit.USER(Ch).NTRansition

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:LIMit:USER[1-1]:PTRansition {0 ~ 65535}

:STATus:QUEStionable:LIMit:USER[1-1]:PTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:LIMit:USER1:PTRansition 0 "
20 OUTPUT 717;":STATus:QUEStionable:LIMit:USER1:PTRansition ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.MISC.ENABle

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.MISC.ENABle = *Value*

Value = SCPI.STATus.QUEStionable.MISC.ENABle

Description

This command sets/gets questionable-misc status enable register.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-misc status enable register
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.MISC.ENABle = Var
Var = SCPI.STATus.QUEStionable.MISC.ENABle
```

Related Objects

SCPI.STATus.QUEStionable.MISC.EVENt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:MISC:ENABle {0 ~ 65535}

:STATus:QUEStionable:MISC:ENABle?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:MISC:ENABle 0 "
20 OUTPUT 717;":STATus:QUEStionable:MISC:ENABle ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.MISC.EVENt

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.MISC.EVENt

Description

This command gets questionable-misc status event register.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-misc status event register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.MISC.EVENt
```

Related Objects

SCPI.STATus.QUEStionable.MISC.ENABle

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:MISC[:EVENt]?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:MISC[:EVENt] ?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.NTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.NTRansition = *Value*

Value = SCPI.STATus.QUEStionable.NTRansition

Description

This command sets/gets questionable status negative transition filter.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable status negative transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.NTRansition = Var
Var = SCPI.STATus.QUEStionable.NTRansition
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:NTRansition {0 ~ 65535}

:STATus:QUEStionable:NTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:NTRansition 0 "
20 OUTPUT 717;":STATus:QUEStionable:NTRansition ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.PHASe.ENABle

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.PHASe.ENABle = *Value*

Value = SCPI.STATus.QUEStionable.PHASe.ENABle

Description

This command sets/gets questionable-phase status enable register.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-phase status enable register
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.PHASe.ENABle = Var
Var = SCPI.STATus.QUEStionable.PHASe.ENABle
```

Related Objects

SCPI.STATus.QUEStionable.PHASe.EVENt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:PHASe:ENABle {0 ~ 65535}

:STATus:QUEStionable:PHASe:ENABle?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:PHASe:ENABle 0 "
20 OUTPUT 717;":STATus:QUEStionable:PHASe:ENABle ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.PHASe.EVENt

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.PHASe.EVENt

Description

This command gets questionable-phase status event register.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-phase status event register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.PHASe.EVENt
```

Related Objects

SCPI.STATus.QUEStionable.PHASe.ENABle

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:PHASe[:EVENt]?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:PHASe[:EVENt] ?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.POWer.ENABle

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.POWer.ENABle = *Value*

Value = SCPI.STATus.QUEStionable.POWer.ENABle

Description

This command sets/gets questionable-power status enable register.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-power status enable register
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.POWer.ENABle = Var
Var = SCPI.STATus.QUEStionable.POWer.ENABle
```

Related Objects

SCPI.STATus.QUEStionable.POWer.EVENt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:POWer:ENABle {0 ~ 65535}

:STATus:QUEStionable:POWer:ENABle?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:POWer:ENABle 0 "
20 OUTPUT 717;":STATus:QUEStionable:POWer:ENABle ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.POWer.EVENt

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.POWer.EVENt

Description

This command gets questionable-power status event register.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-power status event register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.POWer.EVENt
```

Related Objects

SCPI.STATus.QUEStionable.POWer.ENABle

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:POWer[:EVENt]?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:POWer[:EVENt] ?"
20 ENTER 717;A
```

SCPI.STATus.QUEStionable.PTRansition

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.PTRansition = *Value*

Value = SCPI.STATus.QUEStionable.PTRansition

Description

This command sets/gets questionable status positive transition filter.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable status positive transition filter
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	32767
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.PTRansition = Var
Var = SCPI.STATus.QUEStionable.PTRansition
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:PTRansition {0 ~ 65535}

:STATus:QUEStionable:PTRansition?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:PTRansition 0 "
20 OUTPUT 717;":STATus:QUEStionable:PTRansition ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.REFerence.ENABle

Object Type

Property (**Read-Write**)

Syntax

SCPI.STATus.QUEStionable.REFerence.ENABle = *Value*

Value = SCPI.STATus.QUEStionable.REFerence.ENABle

Description

This command sets/gets questionable-reference signal status enable register.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-reference signal status enable register
Data Type	Long integer type (Long)
Range	0 ~ 65535
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
SCPI.STATus.QUEStionable.REFerence.ENABle = Var
Var = SCPI.STATus.QUEStionable.REFerence.ENABle
```

Related Objects

SCPI.STATus.QUEStionable.REFerence.EVENt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:REFerence:ENABle {0 ~ 65535}

:STATus:QUEStionable:REFerence:ENABle?

Query Response

{0 ~ 65535} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:REFerence:ENABle 0 "
20 OUTPUT 717;":STATus:QUEStionable:REFerence:ENABle ?"
30 ENTER 717;A
```

SCPI.STATus.QUEStionable.REFerence.EVENt

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.STATus.QUEStionable.REFerence.EVENt

Description

This command gets questionable-reference signal status event register.

NOTE Effective only for 0-14 bits.

Variable

Parameter	*Value*
Description	Questionable-reference signal status event register
Data Type	Long integer type (Long)
Range	0 ~ 32767
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var= 0
Var = SCPI.STATus.QUEStionable.REFerence.EVENt
```

Related Objects

SCPI.STATus.QUEStionable.REFerence.EVENt

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:STATus:QUEStionable:REFerence[:EVENt]?

Query Response

{0 ~ 32767} <newline><^END>

Example of use

```
10 OUTPUT 717;":STATus:QUEStionable:REFerence[:EVENt] ?"
20 ENTER 717;A
```

System

SCPI.SYSTem.BACKlight.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.SYSTem.BACKlight.STATe = *Value*

Value = SCPI.SYSTem.BACKlight.STATe

Description

This command turns backlight On/Off.

Variable

Parameter	*Value*
Description	Turns backlight On/Off
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.SYSTem.BACKlight.STATe = Var
Var = SCPI.SYSTem.BACKlight.STATe
```

Equivalent Key

PN Menu: **System** > **Backlight**

SP Menu: **System** > **Backlight**

FP Menu: **System** > **Backlight**

TR Menu: **System** > **Backlight**

AM Menu: **System** > **Backlight**

BB Menu: **System** > **Backlight**

USER Menu: **System** > **Backlight**

Equivalent SCPI Command

Syntax

:SYSTem:BACKlight:STATe {ON|OFF|1|0}

:SYSTem:BACKlight:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SYSTem:BACKlight:STATe 1"
20 OUTPUT 717;":SYSTem:BACKlight:STATe ?"
30 ENTER 717;A
```

SCPI.SYSTem.BEEPer.COMPlete.IMMediate

Object Type

Method (**Write Only**)

Syntax

SCPI.SYSTem.BEEPer.COMPlete.IMMediate

Description

This command sets beep complete immediately.

Examples

SCPI.SYSTem.BEEPer.COMPlete.IMMediate

Related Objects

SCPI.SYSTem.BEEPer.COMPlete.STATe

Equivalent Key

PN Menu: **System** > **Misc Setup** > **Beeper** > **Test Beep Complete**

SP Menu: **System** > **Misc Setup** > **Beeper** > **Test Beep Complete**

FP Menu: **System** > **Misc Setup** > **Beeper** > **Test Beep Complete**

TR Menu: **System** > **Misc Setup** > **Beeper** > **Test Beep Complete**

AM Menu: **System** > **Misc Setup** > **Beeper** > **Test Beep Complete**

BB Menu: **System** > **Misc Setup** > **Beeper** > **Test Beep Complete**

USER Menu: **System** > **Misc Setup** > **Beeper** > **Test Beep Complete**

Equivalent SCPI Command

Syntax

:SYSTem:BEEPer:COMPlete:IMMediate

Example of use

```
10 OUTPUT 717;":SYSTem:BEEPer:COMPlete:IMMediate"
```

SCPI.SYSTem.BEEPer.COMPlete.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.SYSTem.BEEPer.COMPlete.STATe = *Value*

Value = SCPI.SYSTem.BEEPer.COMPlete.STATe

Description

This command sets/gets complete beeper state.

Variable

Parameter	*Value*
Description	Complete beeper state
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	ON
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
```

```
SCPI.SYSTem.BEEPer.COMPlete.STATe = Var
Var = SCPI.SYSTem.BEEPer.COMPlete.STATe
```

Related Objects

SCPI.SYSTem.BEEPer.COMPlete.IMMediate

Equivalent Key

PN Menu: **System** > **Misc Setup** > **Beeper** > **Beep Complete**

SP Menu: **System** > **Misc Setup** > **Beeper** > **Beep Complete**

FP Menu: **System** > **Misc Setup** > **Beeper** > **Beep Complete**

TR Menu: **System** > **Misc Setup** > **Beeper** > **Beep Complete**

AM Menu: **System** > **Misc Setup** > **Beeper** > **Beep Complete**

BB Menu: **System** > **Misc Setup** > **Beeper** > **Beep Complete**

USER Menu: **System** > **Misc Setup** > **Beeper** > **Beep Complete**

Equivalent SCPI Command

Syntax

:SYSTem:BEEPer:COMPlete:STATe {ON|OFF|1|0}

:SYSTem:BEEPer:COMPlete:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SYSTem:BEEPer:COMPlete:STATe 1"
20 OUTPUT 717;":SYSTem:BEEPer:COMPlete:STATe ?"
30 ENTER 717;A
```

SCPI.SYSTem.BEEPer.WARNing.IMMediate

Object Type

Method (**Write Only**)

Syntax

SCPI.SYSTem.BEEPer.WARNing.IMMediate

Description

This command sets beep warning immediate.

Examples

SCPI.SYSTem.BEEPer.WARNing.IMMediate

Related Objects

SCPI.SYSTem.BEEPer.WARNing.STATe

Equivalent Key

PN Menu: **System** > **Misc Setup** > **Beeper** > **Test Beep Warning**

SP Menu: **System** > **Misc Setup** > **Beeper** > **Test Beep Warning**

FP Menu: **System** > **Misc Setup** > **Beeper** > **Test Beep Warning**

TR Menu: **System** > **Misc Setup** > **Beeper** > **Test Beep Warning**

AM Menu: **System** > **Misc Setup** > **Beeper** > **Test Beep Warning**

BB Menu: **System** > **Misc Setup** > **Beeper** > **Test Beep Warning**

USER Menu: **System** > **Misc Setup** > **Beeper** > **Test Beep Warning**

Equivalent SCPI Command

Syntax

:SYSTem:BEEPer:WARNing:IMMediate

Example of use

```
10 OUTPUT 717;":SYSTem:BEEPer:WARNing:IMMediate"
```

SCPI.SYSTem.BEEPer.WARNing.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.SYSTem.BEEPer.WARNing.STATe = *Value*

Value = SCPI.SYSTem.BEEPer.WARNing.STATe

Description

This command sets/gets warning beeper state.

Variable

Parameter	*Value*
Description	Warning beeper state
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
```

```
SCPI.SYSTem.BEEPer.WARNing.STATe = Var
Var = SCPI.SYSTem.BEEPer.WARNing.STATe
```

Related Objects

SCPI.SYSTem.BEEPer.WARNing.IMMediate

Equivalent Key

PN Menu: **System** > **Misc Setup** > **Beeper** > **Beep Warning**

SP Menu: **System** > **Misc Setup** > **Beeper** > **Beep Warning**

FP Menu: **System** > **Misc Setup** > **Beeper** > **Beep Warning**

TR Menu: **System** > **Misc Setup** > **Beeper** > **Beep Warning**

AM Menu: **System** > **Misc Setup** > **Beeper** > **Beep Warning**

BB Menu: **System** > **Misc Setup** > **Beeper** > **Beep Warning**

USER Menu: **System** > **Misc Setup** > **Beeper** > **Beep Warning**

Equivalent SCPI Command

Syntax

:SYSTem:BEEPer:WARNing:STATe {ON|OFF|1|0}

:SYSTem:BEEPer:WARNing:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SYSTem:BEEPer:WARNing:STATe 1"
20 OUTPUT 717;":SYSTem:BEEPer:WARNing:STATe ?"
30 ENTER 717;A
```

SCPI.SYSTem.COMMunicate.LAN.CONTrol

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.SYSTem.COMMunicate.LAN.CONTrol

Description

This command gets control port number of socket connection.

Variable

Parameter	*Value*
Description	Control port number of socket connection
Data Type	Long integer type (Long)
Range	-
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Long
Var = SCPI.SYSTem.COMMunicate.LAN.CONTrol
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SYSTem:COMMunicate:LAN:CONTrol?

Query Response

{Long} <newline><^END>

Example of use

```
10 OUTPUT 717;":SYSTem:COMMunicate:LAN:CONTrol ?"
20 ENTER 717;A
```

SCPI.SYSTem.DATE[_Q] year, month, day

Object Type

Property (**Read-Write**)

Syntax

SCPI.SYSTem.DATE[_Q] year, month, day = *Value*

Value = SCPI.SYSTem.DATE[_Q] year, month, day

Description

This command sets/gets system date.

Variable

Parameter	*year*
Description	System date
Data Type	Long integer type (Long)
Range	1980 ~ 2030
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-224, "Illegal parameter value"

Parameter	*month*
Description	System date
Data Type	Long integer type (Long)
Range	1 ~ 12
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-224, "Illegal parameter value"

Parameter	*day*
Description	System date
Data Type	Long integer type (Long)
Range	1 ~ 31
Preset Value	-
Unit	-

Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-224, "Illegal parameter value"

Examples

```
Dim Var as Variant
SCPI.SYSTem.DATE[_Q] year, month, day = Var
Var = SCPI.SYSTem.DATE[_Q] year, month, day
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SYSTem:DATE {1980 ~ 2030 1 ~ 12 1 ~ 31}

:SYSTem:DATE?

Query Response

{1980 ~ 2030 1 ~ 12 1 ~ 31} <newline><^END>

Example of use

```
10 OUTPUT 717;":SYSTem:DATE -"
20 OUTPUT 717;":SYSTem:DATE ?"
20 ENTER 717;A
```

SCPI.SYSTem.ERRor.NEXT_Q err_no, err_desc

Object Type

Property (**Read Only**)

Syntax

Value = SCPI.SYSTem.ERRor.NEXT_Q err_no, err_desc

Description

This command gets error code & description.

Variable

Parameter	*err_no*
Description	Error code
Data Type	Character string type (String)
Range	-
Preset Value	0
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Parameter	*err_desc*
Description	Error description
Data Type	Long integer type (Long)
Range	- 254 chars
Preset Value	"No error"
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Variant
Var = SCPI.SYSTem.ERRor.NEXT_Q err_no, err_desc
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SYSTem:ERRor[:NEXT]?

Query Response

{ - 254 chars} <newline><^END>

Example of use

```
10 OUTPUT 717;":SYSTem:ERRor[:NEXT] ?"
20 ENTER 717;A
```

SCPI.SYSTem.KLOCk.KBD

Object Type

Property (**Read-Write**)

Syntax

SCPI.SYSTem.KLOCk.KBD = *Value*

Value = SCPI.SYSTem.KLOCk.KBD

Description

This command sets/gets front panel and keyboard lock state.

Variable

Parameter	*Value*
Description	Front panel and keyboard lock state
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.SYSTem.KLOCk.KBD = Var
Var = SCPI.SYSTem.KLOCk.KBD
```

Related Objects

SCPI.SYSTem.KLOCk.MOUSe

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SYSTem:KLOCk:KBD {ON|OFF|1|0}

:SYSTem:KLOCk:KBD?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SYSTem:KLOCk:KBD 1"
20 OUTPUT 717;":SYSTem:KLOCk:KBD ?"
30 ENTER 717;A
```

SCPI.SYSTem.KLOCk.MOUSe

Object Type

Property (**Read-Write**)

Syntax

SCPI.SYSTem.KLOCk.MOUSe = *Value*

Value = SCPI.SYSTem.KLOCk.MOUSe

Description

This command sets/gets touch screen and mouse lock state.

Variable

Parameter	*Value*
Description	Touch screen and mouse lock state
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.SYSTem.KLOCk.MOUSe = Var
Var = SCPI.SYSTem.KLOCk.MOUSe
```

Related Objects

SCPI.SYSTem.KLOCk.KBD

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SYSTem:KLOCk:MOUSe {ON|OFF|1|0}

:SYSTem:KLOCk:MOUSe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":SYSTem:KLOCk:MOUSe 1"
20 OUTPUT 717;":SYSTem:KLOCk:MOUSe ?"
30 ENTER 717;A
```

SCPI.SYSTem.POFF

Object Type

Method (**Write Only**)

Syntax

SCPI.SYSTem.POFF

Description

This command turns the instrument power off.

Examples

SCPI.SYSTem.POFF

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SYSTem:POFF

Example of use

```
10 OUTPUT 717;":SYSTem:POFF"
```

SCPI.SYSTem.PRESet

Object Type

Method (**Write Only**)

Syntax

SCPI.SYSTem.PRESet

Description

This command sets preset instrument state. It is same as: '*RST;:INIT:instr:CONT ON'(`instr' is all instrument)..

Examples

SCPI.SYSTem.PRESet

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SYSTem:PRESet

Example of use

10 OUTPUT 717;":SYSTem:PRESet"

SCPI.SYSTem.SECurity.LEVel

Object Type

Property (**Read-Write**)

Syntax

SCPI.SYSTem.SECurity.LEVel = *Value*

Value = SCPI.SYSTem.SECurity.LEVel

Description

This command sets/gets security level setting.

Variable

Parameter	*Value*
Description	Security level setting
Data Type	Character string type (String)
Range	NONE|LOW|HIGH
Preset Value	NONE
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NONE"
SCPI.SYSTem.SECurity.LEVel = Var
Var = SCPI.SYSTem.SECurity.LEVel
```

Equivalent Key

PN Menu: **Display** > **Security Level**

SP Menu: **Display** > **Security Level**

FP Menu: **Display** > **Security Level**

TR Menu: **Display** > **Security Level**

AM Menu: **Display** > **Security Level**

BB Menu: **Display** > **Security Level**

USER Menu: **Display** > **Security Level**

Equivalent SCPI Command

Syntax

:SYSTem:SECurity[:LEVel] {NONE|LOW|HIGH}

:SYSTem:SECurity[:LEVel]?

Query Response

{NONE|LOW|HIGH} <newline><^END>

Example of use

```
10 OUTPUT 717;":SYSTem:SECurity[:LEVel] NONE"
20 OUTPUT 717;":SYSTem:SECurity[:LEVel] ?"
30 ENTER 717;A$
```

SCPI.SYSTem.SET

Object Type

Method (**Write Only**)

Syntax

SCPI.SYSTem.SET

Description

This command sets the instrument to the state determined by the contents of the parameter.

Examples

SCPI.SYSTem.SET

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SYSTem:SET

Example of use

```
10 OUTPUT 717;":SYSTem:SET NONE"
```

SCPI.SYSTem.TIME[_Q] hour, minute, second

Object Type

Property (**Read-Write**)

Syntax

SCPI.SYSTem.TIME[_Q] hour, minute, second = *Value*

Value = SCPI.SYSTem.TIME[_Q] hour, minute, second

Description

This command sets/gets system time.

Variable

Parameter	*hour*
Description	System time
Data Type	Long integer type (Long)
Range	0 ~ 23
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-224, "Illegal parameter value"

Parameter	*minute*
Description	System time
Data Type	Long integer type (Long)
Range	0 ~ 59
Preset Value	-
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-224, "Illegal parameter value"

Parameter	*second*
Description	System time
Data Type	Long integer type (Long)
Range	0 ~ 59
Preset Value	-
Unit	-

Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-224, "Illegal parameter value"

Examples

```
Dim Var as Variant
SCPI.SYSTem.TIME[_Q] hour, minute, second = Var
Var = SCPI.SYSTem.TIME[_Q] hour, minute, second
```

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:SYSTem:TIME {0 ~ 23 0 ~ 59 0 ~ 59}

:SYSTem:TIME?

Query Response

{0 ~ 23 0 ~ 59 0 ~ 59} <newline><^END>

Example of use

```
10 OUTPUT 717;":SYSTem:TIME NONE"
20 OUTPUT 717;":SYSTem:TIME ?"
30 ENTER 717;A$
```

Trigger

SCPI.TRIGger.AM(Ch).SOURce

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.AM(Ch).SOURce = *Value*

Value = SCPI.TRIGger.AM(Ch).SOURce

Description

This command sets/gets trigger source, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trigger source
Data Type	Character string type (String)
Range	INTernal|EXTernal|MANual|BUS
Preset Value	INTernal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "INTernal"
SCPI.TRIGger.AM1.SOURce = Var
Var = SCPI.TRIGger.AM1.SOURce
```

Equivalent Key

AM Menu: **Trigger** > **Source**

Equivalent SCPI Command

Syntax

:TRIGger:AM[1-1]:SOURce {INTernal|EXTernal|MANual|BUS}

:TRIGger:AM[1-1]:SOURce?

Query Response

{INTernal|EXTernal|MANual|BUS} <newline><^END>

Example of use

```
10 OUTPUT 717;":TRIGger:AM1:SOURce INTernal"
20 OUTPUT 717;":TRIGger:AM1:SOURce ?"
30 ENTER 717;A$
```

SCPI.TRIGger.AVERage

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.AVERage = *Value*

Value = SCPI.TRIGger.AVERage

Description

This command sets/gets trigger system initialization average times.

Variable

Parameter	*Value*
Description	Trigger system initialization average times
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.TRIGger.AVERage = Var
Var = SCPI.TRIGger.AVERage
```

Equivalent Key

PN Menu: **Trigger** > **Average Trigger**

SP Menu: **Trigger** > **Average Trigger**

FP Menu: **Trigger** > **Average Trigger**

TR Menu: **Trigger** > **Average Trigger**

AM Menu: **Trigger** > **Average Trigger**

BB Menu: **Trigger** > **Average Trigger**

Equivalent SCPI Command

Syntax

:TRIGger:AVERage {ON|OFF|1|0}

:TRIGger:AVERage?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":TRIGger:AVERage 1"
20 OUTPUT 717;":TRIGger:AVERage ?"
30 ENTER 717;A
```

SCPI.TRIGger.BB(Ch).SOURce

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.BB(Ch).SOURce = *Value*

Value = SCPI.TRIGger.BB(Ch).SOURce

Description

This command sets/gets trigger source, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trigger source
Data Type	Character string type (String)
Range	INTernal|EXTernal|MANual|BUS
Preset Value	INTernal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "INTernal"
SCPI.TRIGger.BB1.SOURce = Var
Var = SCPI.TRIGger.BB1.SOURce
```

Equivalent Key

BB Menu: **Trigger** > **Source**

Equivalent SCPI Command

Syntax

:TRIGger:BB[1-1]:SOURce {INTernal|EXTernal|MANual|BUS}

:TRIGger:BB[1-1]:SOURce?

Query Response

{INTernal|EXTernal|MANual|BUS} <newline><^END>

Example of use

```
10 OUTPUT 717;":TRIGger:BB1:SOURce INTernal"
20 OUTPUT 717;":TRIGger:BB1:SOURce ?"
30 ENTER 717;A$
```

SCPI.TRIGger.EXTernal.SLOPe

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.EXTernal.SLOPe = *Value*

Value = SCPI.TRIGger.EXTernal.SLOPe

Description

This command sets/gets external trigger polarity.

Variable

Parameter	*Value*
Description	External trigger polarity
Data Type	Character string type (String)
Range	NEGative|POSitive
Preset Value	NEGative
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "NEGative"
SCPI.TRIGger.EXTernal.SLOPe = Var
Var = SCPI.TRIGger.EXTernal.SLOPe
```

Equivalent Key

PN Menu: **Trigger** > **Ext Trig Polarity**

SP Menu: **Trigger** > **Ext Trig Polarity**

FP Menu: **Trigger** > **Ext Trig Polarity**

TR Menu: **Trigger** > **Ext Trig Polarity**

AM Menu: **Trigger** > **Ext Trig Polarity**

BB Menu: **Trigger** > **Ext Trig Polarity**

Equivalent SCPI Command

Syntax

:TRIGger:EXTernal:SLOPe {NEGative|POSitive}

:TRIGger:EXTernal:SLOPe?

Query Response

{NEGative|POSitive} <newline><^END>

Example of use

```
10 OUTPUT 717;":TRIGger:EXTernal:SLOPe NEGative"
20 OUTPUT 717;":TRIGger:EXTernal:SLOPe ?"
30 ENTER 717;A$
```

SCPI.TRIGger.FP(Ch).MODE

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.FP(Ch).MODE = *Value*

Value = SCPI.TRIGger.FP(Ch).MODE

Description

This command sets/gets trigger mode, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trigger mode
Data Type	Character string type (String)
Range	ANALyzer|TESTer
Preset Value	ANALyzer
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	100, "Option not installed"

Examples

```
Dim Var as String
Var= "ANALyzer"
SCPI.TRIGger.FP1.MODE = Var
Var = SCPI.TRIGger.FP1.MODE
```

Related Objects

SCPI.TRIGger.FP(Ch).SOURce

Equivalent Key

FP Menu: **Trigger** > **Mode**

Equivalent SCPI Command

Syntax

:TRIGger:FP[1-1]:MODE {ANALyzer|TESTer}

:TRIGger:FP[1-1]:MODE?

Query Response

{ANALyzer|TESTer} <newline><^END>

Example of use

```
10 OUTPUT 717;":TRIGger:FP1:MODE ANALyzer"
20 OUTPUT 717;":TRIGger:FP1:MODE ?"
30 ENTER 717;A$
```

SCPI.TRIGger.FP(Ch).SOURce

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.FP(Ch).SOURce = *Value*

Value = SCPI.TRIGger.FP(Ch).SOURce

Description

This command sets/gets trigger source, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trigger source
Data Type	Character string type (String)
Range	INTernal|EXTernal|MANual|BUS
Preset Value	INTernal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "INTernal"
SCPI.TRIGger.FP1.SOURce = Var
Var = SCPI.TRIGger.FP1.SOURce
```

Related Objects

SCPI.TRIGger.FP(Ch).MODE

Equivalent Key

FP Menu: **Trigger** > **Source**

Equivalent SCPI Command

Syntax

:TRIGger:FP[1-1]:SOURce {INTernal|EXTernal|MANual|BUS}

:TRIGger:FP[1-1]:SOURce?

Query Response

{INTernal|EXTernal|MANual|BUS} <newline><^END>

Example of use

```
10 OUTPUT 717;":TRIGger:FP1:SOURce INTernal"
20 OUTPUT 717;":TRIGger:FP1:SOURce ?"
30 ENTER 717;A$
```

SCPI.TRIGger.MODE

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.MODE = *Value*

Value = SCPI.TRIGger.MODE

Description

This command sets/gets measurement mode.

Variable

Parameter	*Value*
Description	Measurement mode
Data Type	Character string type (String)
Range	PN1|SP1|FP1|TR1|AM1|BB1|PS1
Preset Value	PN1
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	100, "Option not installed"

Examples

```
Dim Var as String
Var= "PN1"
SCPI.TRIGger.MODE = Var
Var = SCPI.TRIGger.MODE
```

Equivalent Key

PN Menu: **Trigger** > **Trigger to Phase Noise**

SP Menu: **Trigger** > **Trigger to Spectrum Monitor**

FP Menu: **Trigger** > **Trigger to Freq & Power**

TR Menu: **Trigger** > **Trigger to Transient**

AM Menu: **Trigger** > **Trigger to AM Noise**

BB Menu: **Trigger** > **Trigger to Baseband Noise**

PS Menu: **Trigger** > **Trigger to Segment PN**

Equivalent SCPI Command

Syntax

:TRIGger:MODE {PN1|SP1|FP1|TR1|AM1|BB1|PS1}

:TRIGger:MODE?

Query Response

{PN1|SP1|FP1|TR1|AM1|BB1|PS1} <newline><^END>

Example of use

```
10 OUTPUT 717;":TRIGger:MODE PN1"
20 OUTPUT 717;":TRIGger:MODE ?"
30 ENTER 717;A$
```

SCPI.TRIGger.OUTPut.POLarity

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.OUTPut.POLarity = *Value*

Value = SCPI.TRIGger.OUTPut.POLarity

Description

This command sets/gets external trigger output polarity.

Variable

Parameter	*Value*
Description	External trigger output polarity
Data Type	Character string type (String)
Range	POSitive|NEGative
Preset Value	POSitive
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "POSitive"
SCPI.TRIGger.OUTPut.POLarity = Var
Var = SCPI.TRIGger.OUTPut.POLarity
```

Related Objects

SCPI.TRIGger.OUTPut.POLarity

SCPI.TRIGger.OUTPut.SELect

SCPI.TRIGger.OUTPut.STATe

Equivalent Key

PN Menu: **Trigger** > **Ext Trig Output** > **Polarity**

SP Menu: **Trigger** > **Ext Trig Output** > **Polarity**

FP Menu: **Trigger** > **Ext Trig Output** > **Polarity**

TR Menu: **Trigger** > **Ext Trig Output** > **Polarity**

AM Menu: **Trigger** > **Ext Trig Output** > **Polarity**

BB Menu: **Trigger** > **Ext Trig Output** > **Polarity**

Equivalent SCPI Command

Syntax

:TRIGger:OUTPut:POLarity {POSitive|NEGative}

:TRIGger:OUTPut:POLarity?

Query Response

{POSitive|NEGative} <newline><^END>

Example of use

```
10 OUTPUT 717;":TRIGger:OUTPut:POLarity POSitive"
20 OUTPUT 717;":TRIGger:OUTPut:POLarity ?"
30 ENTER 717;A$
```

SCPI.TRIGger.OUTPut.SELect

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.OUTPut.SELect = *Value*

Value = SCPI.TRIGger.OUTPut.SELect

Description

This command sets/gets external trigger output timing selection.

Variable

Parameter	*Value*
Description	External trigger output timing selection
Data Type	Character string type (String)
Range	EOM|INDex
Preset Value	EOM
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "EOM"
SCPI.TRIGger.OUTPut.SELect = Var
Var = SCPI.TRIGger.OUTPut.SELect
```

Related Objects

SCPI.TRIGger.OUTPut.POLarity

SCPI.TRIGger.OUTPut.STATe

Equivalent Key

PN Menu: **Trigger** > **Ext Trig Output** > **Selection**

SP Menu: **Trigger** > **Ext Trig Output** > **Selection**

FP Menu: **Trigger** > **Ext Trig Output** > **Selection**

TR Menu: **Trigger** > **Ext Trig Output** > **Selection**

AM Menu: **Trigger** > **Ext Trig Output** > **Selection**

BB Menu: **Trigger** > **Ext Trig Output** > **Selection**

Equivalent SCPI Command

Syntax

:TRIGger:OUTPut:SELect {EOM|INDex}

:TRIGger:OUTPut:SELect?

Query Response

{EOM|INDex} <newline><^END>

Example of use

```
10 OUTPUT 717;":TRIGger:OUTPut:SELect EOM"
20 OUTPUT 717;":TRIGger:OUTPut:SELect ?"
30 ENTER 717;A$
```

SCPI.TRIGger.OUTPut.STATe

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.OUTPut.STATe = *Value*

Value = SCPI.TRIGger.OUTPut.STATe

Description

This command sets/gets External trigger output ON/OFF.

Variable

Parameter	*Value*
Description	External trigger output (On/Off)
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.TRIGger.OUTPut.STATe = Var
Var = SCPI.TRIGger.OUTPut.STATe
```

Related Objects

SCPI.TRIGger.OUTPut.POLarity

SCPI.TRIGger.OUTPut.SELect

Equivalent Key

PN Menu: **Trigger** > **Ext Trig Output** > **State**

SP Menu: **Trigger** > **Ext Trig Output** > **State**

FP Menu: **Trigger** > **Ext Trig Output** > **State**

TR Menu: **Trigger** > **Ext Trig Output** > **State**

AM Menu: **Trigger** > **Ext Trig Output** > **State**

BB Menu: **Trigger** > **Ext Trig Output** > **State**

Equivalent SCPI Command

Syntax

:TRIGger:OUTPut:STATe {ON|OFF|1|0}

:TRIGger:OUTPut:STATe?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":TRIGger:OUTPut:STATe 1"
20 OUTPUT 717;":TRIGger:OUTPut:STATe ?"
30 ENTER 717;A
```

SCPI.TRIGger.PN(Ch).SOURce

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.PN(Ch).SOURce = *Value*

Value = SCPI.TRIGger.PN(Ch).SOURce

Description

This command sets/gets trigger source, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trigger source
Data Type	Character string type (String)
Range	INTernal|EXTernal|MANual|BUS
Preset Value	INTernal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "INTernal"
SCPI.TRIGger.PN1.SOURce = Var
Var = SCPI.TRIGger.PN1.SOURce
```

Equivalent Key

PN Menu: **Trigger** > **Source**

Equivalent SCPI Command

Syntax

:TRIGger:PN[1-1]:SOURce {INTernal|EXTernal|MANual|BUS}

:TRIGger:PN[1-1]:SOURce?

Query Response

{INTernal|EXTernal|MANual|BUS} <newline><^END>

Example of use

```
10 OUTPUT 717;":TRIGger:PN1:SOURce INTernal"
20 OUTPUT 717;":TRIGger:PN1:SOURce ?"
30 ENTER 717;A$
```

SCPI.TRIGger.PS(Ch).SOURce

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.PS(Ch).SOURce = *Value*

Value = SCPI.TRIGger.PS(Ch).SOURce

Description

This command sets/gets trigger source, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trigger source
Data Type	Character string type (String)
Range	INTernal|EXTernal|MANual|BUS
Preset Value	INTernal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "INTernal"
SCPI.TRIGger.PS1.SOURce = Var
Var = SCPI.TRIGger.PS1.SOURce
```

Equivalent Key

PS Menu: **Trigger** > **Source**

Equivalent SCPI Command

Syntax

:TRIGger:PS[1-1]:SOURce {INTernal|EXTernal|MANual|BUS}

:TRIGger:PS[1-1]:SOURce?

Query Response

{INTernal|EXTernal|MANual|BUS} <newline><^END>

Example of use

```
10 OUTPUT 717;":TRIGger:PS1:SOURce INTernal"
20 OUTPUT 717;":TRIGger:PS1:SOURce?"
30 ENTER 717;A$
```

SCPI.TRIGger.SOPC

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.SOPC = *Value*

Value = SCPI.TRIGger.SOPC

Description

This command sets/gets Trigger system synchronous to *OPC?/*OPC/*WAI.

Variable

Parameter	*Value*
Description	Trigger system synchronous to *OPC?/*OPC/*WAI
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	No
Save-Recall Effect	No
Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.TRIGger.SOPC = Var
Var = SCPI.TRIGger.SOPC
```

Equivalent Key

PN Menu: **System** > **Service Menu** > **Service Function** > **Service Functions** > **Misc** > **Trigger Sync. OPC**

SP Menu: **System** > **Service Menu** > **Service Function** > **Service Functions** > **Misc** > **Trigger Sync. OPC**

FP Menu: **System** > **Service Menu** > **Service Function** > **Service Functions** > **Misc** > **Trigger Sync. OPC**

TR Menu: **System** > **Service Menu** > **Service Function** > **Service Functions** > **Misc** > **Trigger Sync. OPC**

AM Menu: **System** > **Service Menu** > **Service Function** > **Service Functions** > **Misc** > **Trigger Sync. OPC**

BB Menu: **System** > **Service Menu** > **Service Function** > **Service Functions** > **Misc** > **Trigger Sync. OPC**

USER Menu: **System** > **Service Menu** > **Service Function** > **Service Functions** > **Misc** > **Trigger Sync. OPC**

Equivalent SCPI Command

Syntax

:TRIGger:SOPC {ON|OFF|1|0}

:TRIGger:SOPC?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":TRIGger:SOPC 1"
20 OUTPUT 717;":TRIGger:SOPC ?"
30 ENTER 717;A
```

SCPI.TRIGger.SP(Ch).SOURce

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.SP(Ch).SOURce = *Value*

Value = SCPI.TRIGger.SP(Ch).SOURce

Description

This command sets/gets trigger source, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Trigger source
Data Type	Character string type (String)
Range	INTernal|EXTernal|MANual|BUS
Preset Value	INTernal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "INTernal"
```

```
SCPI.TRIGger.SP1.SOURce = Var
Var = SCPI.TRIGger.SP1.SOURce
```

Equivalent Key

SP Menu: **Trigger** > **Source**

Equivalent SCPI Command

Syntax

:TRIGger:SP[1-1]:SOURce {INTernal|EXTernal|MANual|BUS}

:TRIGger:SP[1-1]:SOURce?

Query Response

{INTernal|EXTernal|MANual|BUS} <newline><^END>

Example of use

```
10 OUTPUT 717;":TRIGger:SP1:SOURce INTernal"
20 OUTPUT 717;":TRIGger:SP1:SOURce ?"
30 ENTER 717;A$
```

SCPI.TRIGger.TR(Ch).ETTAdjust

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.TR(Ch).ETTAdjust = *Value*

Value = SCPI.TRIGger.TR(Ch).ETTAdjust

Description

This command sets/gets external trigger timing adjust, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	External trigger timing adjust
Data Type	Double precision floating point type (Double)
Range	0 ~ 1u
Preset Value	0
Unit	s
Resolution	10n
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.TRIGger.TR1.ETTAdjust = Var
Var = SCPI.TRIGger.TR1.ETTAdjust
```

Equivalent Key

TR Menu: **Trigger** > **Ext Trig Timing Adj.**

Equivalent SCPI Command

Syntax

```
:TRIGger:TR[1-1]:ETTAdjust {0 ~ 1u}
:TRIGger:TR[1-1]:ETTAdjust?
```

Query Response

```
{0 ~ 1u} <newline><^END>
```

Example of use

```
10 OUTPUT 717;":TRIGger:TR1:ETTAdjust 0"
20 OUTPUT 717;":TRIGger:TR1:ETTAdjust ?"
30 ENTER 717;A
```

SCPI.TRIGger.TR(Ch).NARRow(Nr).VIDeo.FREQuency.CENTer

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.TR(Ch).NARRow(Nr).VIDeo.FREQuency.CENTer = *Value*

Value = SCPI.TRIGger.TR(Ch).NARRow(Nr).VIDeo.FREQuency.CENTer

Description

This command sets/gets narrow video trigger frequency, for the selected narrow *Nr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Narrow video trigger frequency
Data Type	Double precision floating point type (Double)
Range	9.2M ~ 337.04G
Preset Value	1.00390625G
Unit	Hz
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 9200000
SCPI.TRIGger.TR1.NARRow1.VIDeo.FREQuency.CENTer = Var
Var = SCPI.TRIGger.TR1.NARRow1.VIDeo.FREQuency.CENTer
```

Related Objects

SCPI.TRIGger.TR(Ch).NARRow(Nr).VIDeo.FREQuency.WIDTh

Equivalent Key

TR Menu: **Setup** > **Video Trigger** > **Narrow Freq**

TR Menu: **Setup** > **Video Trigger** > **Narrow2 Freq**

Equivalent SCPI Command

Syntax

:TRIGger:TR[1-1]:NARRow[1-2]:VIDeo:FREQuency:CENTer {9.2M ~ 337.04G}

:TRIGger:TR[1-1]:NARRow[1-2]:VIDeo:FREQuency:CENTer?

Query Response

{9.2M ~ 337.04G} <newline><^END>

Example of use

```
10 OUTPUT 717;":TRIGger:TR1:NARRow1:VIDeo:FREQuency:CENTer 9200000"
20 OUTPUT 717;":TRIGger:TR1:NARRow1:VIDeo:FREQuency:CENTer ?"
30 ENTER 717;A
```

SCPI.TRIGger.TR(Ch).NARRow(Nr).VIDeo.FREQuency.WIDTh

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.TR(Ch).NARRow(Nr).VIDeo.FREQuency.WIDTh = *Value*

Value = SCPI.TRIGger.TR(Ch).NARRow(Nr).VIDeo.FREQuency.WIDTh

Description

This command sets/gets narrow video trigger frequency span, for the selected narrow *Nr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Narrow video trigger frequency span
Data Type	Double precision floating point type (Double)
Range	0 ~ 85M
Preset Value	13.8M
Unit	Hz
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.TRIGger.TR1.NARRow1.VIDeo.FREQuency.WIDTh = Var
Var = SCPI.TRIGger.TR1.NARRow1.VIDeo.FREQuency.WIDTh
```

Related Objects

SCPI.TRIGger.TR(Ch).NARRow(Nr).VIDeo.FREQuency.CENTer

Equivalent Key

TR Menu: **Setup** > **Video Trigger** > **Narrow Width**

TR Menu: **Setup** > **Video Trigger** > **Narrow2 Width**

Equivalent SCPI Command

Syntax

:TRIGger:TR[1-1]:NARRow[1-2]:VIDeo:FREQuency:WIDTh {0 ~ 85M}

:TRIGger:TR[1-1]:NARRow[1-2]:VIDeo:FREQuency:WIDTh?

Query Response

{0 ~ 85M} <newline><^END>

Example of use

```
10 OUTPUT 717;":TRIGger:TR1:NARRow1:VIDeo:FREQuency:WIDTh 0"
20 OUTPUT 717;":TRIGger:TR1:NARRow1:VIDeo:FREQuency:WIDTh ?"
30 ENTER 717;A
```

SCPI.TRIGger.TR(Ch).NARRow(Nr).VIDeo.MODE

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.TR(Ch).NARRow(Nr).VIDeo.MODE = *Value*

Value = SCPI.TRIGger.TR(Ch).NARRow(Nr).VIDeo.MODE

Description

This command sets/gets narrow video trigger mode, for the selected narrow *Nr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Narrow video trigger mode
Data Type	Character string type (String)
Range	IN|OUT|POSitive|NEGative
Preset Value	IN
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "IN"
SCPI.TRIGger.TR1.NARRow1.VIDeo.MODE = Var
Var = SCPI.TRIGger.TR1.NARRow1.VIDeo.MODE
```

Equivalent Key

TR Menu: **Setup** > **Video Trigger** > **Narrow Video Mode**

TR Menu: **Setup** > **Video Trigger** > **Narrow2 Video Mode**

Equivalent SCPI Command

Syntax

:TRIGger:TR[1-1]:NARRow[1-2]:VIDeo:MODE {IN|OUT|POSitive|NEGative}

:TRIGger:TR[1-1]:NARRow[1-2]:VIDeo:MODE?

Query Response

{IN|OUT|POSitive|NEGative} <newline><^END>

Example of use

```
10 OUTPUT 717;":TRIGger:TR1:NARRow1:VIDeo:MODE IN"
20 OUTPUT 717;":TRIGger:TR1:NARRow1:VIDeo:MODE ?"
30 ENTER 717;A$
```

SCPI.TRIGger.TR(Ch).NARRow(Nr).VIDeo.THReshold

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.TR(Ch).NARRow(Nr).VIDeo.THReshold = *Value*

Value = SCPI.TRIGger.TR(Ch).NARRow(Nr).VIDeo.THReshold

Description

This command sets/gets video trigger threshold level relative to max input level, for the selected narrow *Nr* of the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Video trigger threshold level relative to max input level
Data Type	Double precision floating point type (Double)
Range	-100 ~ 0
Preset Value	-20
Unit	dB
Resolution	1
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.TRIGger.TR1.NARRow1.VIDeo.THReshold = Var
Var = SCPI.TRIGger.TR1.NARRow1.VIDeo.THReshold
```

Equivalent Key

TR Menu: **Setup** > **Video Trigger** > **Minimum Power Level**

Equivalent SCPI Command

Syntax

:TRIGger:TR[1-1]:NARRow[1-2]:VIDeo:THReshold {-100 ~ 0}

:TRIGger:TR[1-1]:NARRow[1-2]:VIDeo:THReshold?

Query Response

{-100 ~ 0} <newline><^END>

Example of use

```
10 OUTPUT 717;":TRIGger:TR1:NARRow1:VIDeo:THReshold 0"
20 OUTPUT 717;":TRIGger:TR1:NARRow1:VIDeo:THReshold ?"
30 ENTER 717;A
```

SCPI.TRIGger.TR(Ch).SOURce

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.TR(Ch).SOURce = *Value*

Value = SCPI.TRIGger.TR(Ch).SOURce

Description

This command sets/gets trigger source, for the selected channel *Ch*.

NOTE Sets N2Video automatically when the measurement mode changed from WN to NN if selected WVIDeo.

NOTE Sets WVIDeo automatically when the measurement mode changed form NN to WN if selected N2Video.

Variable

Parameter	*Value*
Description	Trigger source
Data Type	Character string type (String)
Range	INTernal|EXTernal|MANual|BUS|WVIDeo|NVIDeo|N2Video
Preset Value	INTernal
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as String
Var= "INTernal"
SCPI.TRIGger.TR1.SOURce = Var
Var = SCPI.TRIGger.TR1.SOURce
```

Equivalent Key

TR Menu: **Trigger** > **Source**

Equivalent SCPI Command

Syntax

:TRIGger:TR[1-1]:SOURce
{INTernal|EXTernal|MANual|BUS|WVIDeo|NVIDeo|N2Video}

:TRIGger:TR[1-1]:SOURce?

Query Response

{INTernal|EXTernal|MANual|BUS|WVIDeo|NVIDeo|N2Video} <newline><^END>

Example of use

```
10 OUTPUT 717;":TRIGger:TR1:SOURce INTernal"
20 OUTPUT 717;":TRIGger:TR1:SOURce ?"
30 ENTER 717;A$
```

SCPI.TRIGger.TR(Ch).WIDE.VIDeo.FREQuency.CENTer

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.TR(Ch).WIDE.VIDeo.FREQuency.CENTer = *Value*

Value = SCPI.TRIGger.TR(Ch).WIDE.VIDeo.FREQuency.CENTer

Description

This command sets/gets wide video trigger frequency, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Wide video trigger frequency
Data Type	Double precision floating point type (Double)
Range	50M ~ 337G
Preset Value	1G
Unit	Hz
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 50000000
SCPI.TRIGger.TR1.WIDE.VIDeo.FREQuency.CENTer = Var
Var = SCPI.TRIGger.TR1.WIDE.VIDeo.FREQuency.CENTer
```

Related Objects

SCPI.TRIGger.TR(Ch).WIDE.VIDeo.FREQuency.WIDTh

Equivalent Key

TR Menu: **Setup** > **Video Trigger** > **Wide Freq**

Equivalent SCPI Command

Syntax

:TRIGger:TR[1-1]:WIDE:VIDeo:FREQuency:CENTer {50M ~ 337G}

:TRIGger:TR[1-1]:WIDE:VIDeo:FREQuency:CENTer?

Query Response

{50M ~ 337G} <newline><^END>

Example of use

```
10 OUTPUT 717;":TRIGger:TR1:WIDE:VIDeo:FREQuency:CENTer 50000000"
20 OUTPUT 717;":TRIGger:TR1:WIDE:VIDeo:FREQuency:CENTer ?"
30 ENTER 717;A
```

SCPI.TRIGger.TR(Ch).WIDE.VIDeo.FREQuency.WIDTh

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.TR(Ch).WIDE.VIDeo.FREQuency.WIDTh = *Value*

Value = SCPI.TRIGger.TR(Ch).WIDE.VIDeo.FREQuency.WIDTh

Description

This command sets/gets wide video trigger frequency span, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Wide video trigger frequency span
Data Type	Double precision floating point type (Double)
Range	0 ~ 5.28G
Preset Value	880M
Unit	Hz
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as Double
Var= 0
SCPI.TRIGger.TR1.WIDE.VIDeo.FREQuency.WIDTh = Var
Var = SCPI.TRIGger.TR1.WIDE.VIDeo.FREQuency.WIDTh
```

Related Objects

SCPI.TRIGger.TR(Ch).WIDE.VIDeo.FREQuency.CENTer

Equivalent Key

No equivalent key is available on the front panel.

Equivalent SCPI Command

Syntax

:TRIGger:TR[1-1]:WIDE:VIDeo:FREQuency:WIDTh {0 ~ 5.28G}

:TRIGger:TR[1-1]:WIDE:VIDeo:FREQuency:WIDTh?

Query Response

{0 ~ 5.28G} <newline><^END>

Example of use

```
10 OUTPUT 717;":TRIGger:TR1:WIDE:VIDeo:FREQuency:WIDTh 0"
20 OUTPUT 717;":TRIGger:TR1:WIDE:VIDeo:FREQuency:WIDTh ?"
30 ENTER 717;A
```

SCPI.TRIGger.TR(Ch).WIDE.VIDeo.MODE

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.TR(Ch).WIDE.VIDeo.MODE = *Value*

Value = SCPI.TRIGger.TR(Ch).WIDE.VIDeo.MODE

Description

This command sets/gets wide video trigger mode, for the selected channel *Ch*.

Variable

Parameter	*Value*
Description	Wide video trigger mode
Data Type	Character string type (String)
Range	IN|OUT|POSitive|NEGative
Preset Value	IN
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes
Error	-

Examples

```
Dim Var as String
Var= "IN"
SCPI.TRIGger.TR1.WIDE.VIDeo.MODE = Var
Var = SCPI.TRIGger.TR1.WIDE.VIDeo.MODE
```

Equivalent Key

TR Menu: **Setup** > **Video Trigger** > **Wide Video Mode**

Equivalent SCPI Command

Syntax

:TRIGger:TR[1-1]:WIDE:VIDeo:MODE {IN|OUT|POSitive|NEGative}

:TRIGger:TR[1-1]:WIDE:VIDeo:MODE?

Query Response

{IN|OUT|POSitive|NEGative} <newline><^END>

Example of use

```
10 OUTPUT 717;":TRIGger:TR1:WIDE:VIDeo:MODE IN"
20 OUTPUT 717;":TRIGger:TR1:WIDE:VIDeo:MODE ?"
30 ENTER 717;A$
```

SCPI.TRIGger.WCOuple

Object Type

Property (**Read-Write**)

Syntax

SCPI.TRIGger.WCOuple = *Value*

Value = SCPI.TRIGger.WCOuple

Description

This command sets/gets couple active window and trigger mode.

NOTE This function cannot be used if you changed active window to USER with couple ON.

NOTE The active window remains even after changing the trigger mode with couple ON.

Variable

Parameter	*Value*
Description	Couple active window and trigger mode
Data Type	Boolean type (Boolean)
Range	ON|OFF|1|0
Preset Value	OFF
Unit	-
Resolution	-
Preset Effect	Yes
Save-Recall Effect	Yes

Error	-

Examples

```
Dim Var as Boolean
Var= 1
SCPI.TRIGger.WCOuple = Var
Var = SCPI.TRIGger.WCOuple
```

Equivalent Key

PN Menu: **Trigger** > **Window Couple**

SP Menu: **Trigger** > **Window Couple**

FP Menu: **Trigger** > **Window Couple**

TR Menu: **Trigger** > **Window Couple**

AM Menu: **Trigger** > **Window Couple**

PS Menu: **Trigger** > **Window Couple**

BB Menu: **Trigger** > **Window Couple**

Equivalent SCPI Command

Syntax

:TRIGger:WCOuple {ON|OFF|1|0}

:TRIGger:WCOuple?

Query Response

{ON|OFF|1|0} <newline><^END>

Example of use

```
10 OUTPUT 717;":TRIGger:WCOuple 1"
20 OUTPUT 717;":TRIGger:WCOuple?"
30 ENTER 717;A
```