Programming Manual

VNA Master™ Programming Manual

MS20xxB

MS2024B Vector Network Analyzer

MS2025B Vector Network Analyzer

MS2034B Vector Network Analyzer and Spectrum Analyzer

MS2035B Vector Network Analyzer and Spectrum Analyzer

Anritsu Company
490 Jarvis Drive
Morgan Hill, CA 95037-2809
USA
http://www.anritsu.com

Part Number: 10580-00302
Revision: L
Published: August 2026
Copyright 2026 Anritsu Company

Table of Contents

Chapter 1—General Information

1-1 About this Manual 1-1

1-2 Introduction 1-1

1-3 Remote Programming Setup and Interface 1-2
- USB Interface Connection and Setup 1-2
- Ethernet Interface Connection and Setup 1-4
- Connectivity 1-6

1-4 Sending SCPI Commands 1-7

Chapter 2—Programming with SCPI

2-1 Introduction 2-1

2-2 Introduction to SCPI Programming 2-1

2-3 SCPI Common Commands 2-2

2-4 SCPI Required Commands 2-3

2-5 SCPI Optional Commands 2-3

2-6 Subsystem Commands 2-4
- Command Names 2-4
- Hierarchical Command Structure 2-5
- Query Commands 2-7
- Identifiers 2-7
- Data Parameters 2-8
- Data Parameter Notations 2-9
- Unit Suffixes 2-9

2-7 Notational Conventions 2-10

2-8 Notational Examples 2-11
- Command Terminators 2-11

2-9 Formatting Conventions 2-12

2-10 Command and Query Notational Conventions 2-12

2-11 Parameter Names 2-12

Chapter 3—VNA Commands

3-1 Introduction 3-1
- VNA Commands 3-1

3-2 :CALCulate Subsystem 3-2
- Trace Data Transfer 3-3

3-3 :CALCulate:FORMat Subsystem 3-5
Graph Type 3-5
3-4 :CALCulate:LIMit Subsystem 3-6
Limit Alarm 3-6
Number of Lower Limit Points 3-6
Add Lower Limit Point 3-7
Delete Lower Limit Point 3-7
Lower Limit Next Point Left 3-8
Lower Limit Next Point Right 3-8
Lower Limit Point X Value 3-9
Lower Limit Point Y Value 3-10
Lower Limit State 3-11
Lower Limit X Value 3-12
Lower Limit Y Value 3-13
Number of Limit Points 3-13
Add Limit Point 3-14
Delete Limit Point 3-14
Next Point Left 3-15
Next Point Right 3-15
Limit Point X Value 3-16
Limit Point Y Value 3-17
Limit State 3-18
Limit Type 3-19
Number of Upper Limit Points 3-20
Add Upper Limit Point 3-20
Delete Upper Limit Point 3-21
Upper Limit Next Point Left 3-21
Upper Limit Next Point Right 3-22
Upper Limit Point X Value 3-23
Upper Limit Point Y Value 3-24
Upper Limit State 3-25
Upper Limit X Value 3-26
Upper Limit Y Value 3-27
Limit X Value 3-28
Limit Y Value 3-29

3-5 :CALCulate:MARKer Subsystem 3-30
Turn All Markers Off 3-30
Marker Data 3-30
Delta Marker Reference To 3-31
Delta Marker State 3-32
Marker Readout Format 3-33
Marker Domain Type 3-33
Marker Readout Style 3-34
Marker (Maximum) Peak Search 3-35
Marker (Minimum) Valley Search 3-35
Reference Marker State 3-36
Marker On Trace 3-37
Marker Type 3-38
Marker X Value 3-39
Marker Read Y Value 3-40
3-6 :CALCulate:MATH Subsystem 3-42
Trace To Memory 3-42
3-7 :CALCulate:SMOothing Subsystem 3-43
Smoothing 3-43
3-8 :CALCulate:TRANsform Subsystem 3-44
Maximum Distance 3-44
Distance Resolution 3-44
Start Distance 3-45
Stop Distance 3-45
Distance Units 3-46
Get Distance List 3-46
3-9 :Display Subsystem 3-47
Trace Display 3-47
Trace Format 3-47
Group Delay Aperture 3-48
Scale Resolution Per Division 3-49
Scale Reference Level 3-50
Scale Reference Line 3-51
Smith Chart Scalable Type 3-52
3-10 :Format Subsystem 3-53
Numeric Data Format 3-53
3-11 :INITiate Subsystem 3-54
Continuous/Single Sweep 3-54
Hold Sweep 3-55
Trigger Sweep/Measurement 3-55

3-12 :INPut Subsystem 3-56
Internal Bias Tee Current. 3-56
Internal Bias Tee Tripped State 3-56
Bias Tee State 3-57
Internal Bias Tee Voltage 3-57
3-13 :MMEMory Subsystem 3-58
3-14 :MMEMory:LOAD Subsystem 3-59
Recall Setup 3-59
Recall Measurement 3-60
3-15 :MMEMory:STORe Subsystem 3-61
Save Setup 3-61
Save Measurement 3-62
3-16 [:SENSe] Subsystem 3-63
RF Power In Hold State 3-63
3-17 [:SENSe]:APPLication Subsystem 3-64
Application Self Test 3-64
Application Self Test Result 3-64
3-18 [:SENSe]:AVERage Subsystem 3-65
Restart Averaging 3-65
Number of Traces to Average 3-65
3-19 [:SENSe]:CALibration Subsystem. 3-66
Calibration State 3-66
3-20 [:SENSe]:CORRection Subsystem 3-67
Error Correction Data. 3-68
Smith Chart Reference Impedance 3-69
Calibration Correction State. 3-69
3-21 [:SENSe]:CORRection:CKIT Subsystem 3-70
Calibration Connector Information 3-70
3-22 [:SENSe]:CORRection:CKIT:USER Subsystem 3-75
DUT User Inductance Coefficient value. 3-75
DUT User Capacitance Coefficient value 3-75
DUT User Name 3-76
DUT User Open Offset 3-76
DUT User Short Offset for calibration method SOLT. 3-77

3-23 [:SENSe]:CORRection:COLLect Subsystem 3-78
Calibration Abort 3-78
Calibration Steps 3-79
Calibration Steps and Calibration Types: 3-80
Calibration Step Status 3-81
DUT Port Setup 3-81
Calibration Interpolation. 3-85
Calibration Line Type. 3-85
Calibration Method 3-86
Calculate Calibration Data. 3-86
Calibration Status 3-86
Calibration Accuracy Status 3-87
Calibration Type 3-88
3-24 [:SENSe]:DITHering 3-89
3-25 [:SENSe]:FREQuency Subsystem 3-90
Center Frequency 3-90
Frequency Span 3-91
Distance Suggested Frequency Span 3-91
Start Frequency 3-92
Stop Frequency 3-93
Get Frequency List 3-94
3-26 [:SENSe]:SWEep Subsystem 3-95
IF Bandwidth 3-95
Number of Sweep Points. 3-96
Sweep Type. 3-96
3-27 [:SENSe]:TRACe Subsystem 3-97
Trace Domain 3-97
Trace Select 3-97
S Parameter 3-98
Number of Traces 3-98
Active Trace. 3-99
3-28 :SOURce Subsystem 3-100
Power Levels 3-100
3-29 :SOURce:CORRection:RVELocity Subsystem. 3-101
Propagation Velocity 3-101
Cable Loss. 3-101
3-30 :STATus Subsystem 3-102
Query Operation Status. 3-102

3-31 :SYSTem Subsystem 3-103
System Alternate State 3-103
Motherboard Temperature 3-103
Preset 3-104
3-32 :Trace Subsystem 3-105
Trace Data Transfer 3-105
Trace Header Transfer 3-106

Chapter 4—Vector Voltmeter Commands

4-1 Introduction 4-1
4-2 Vector Voltmeter Commands 4-1
4-3 VVM Commands 4-2
4-4 :MMEMory Subsystem 4-3
4-5 :MMEMory:LOAD Subsystem 4-4
Recall Setup 4-4
Recall Measurement 4-5
4-6 :MMEMory:STORe Subsystem 4-6
Save Setup 4-6
Save Measurement 4-7
4-7 :TRACe VVM Subsystem 4-8
Trace Header Transfer 4-8
Trace Header Parameters 4-9
4-8 [:SENSe]:SWEep Subsystem 4-11
IF Bandwidth 4-11
4-9 [:SENSe]:VVM Subsystem 4-12
Return Measurement Format 4-12
Measurement Mode 4-13
Measurement Type 4-13
4-10 [:SENSe:]:VVM:CABLe Subsystem 4-14
Cable 4-14
4-11 [:SENSe]:VVM:FREQuency Subsystem 4-15
CW Frequency 4-15
4-12 [:SENSe:]:VVM:REFerence Subsystem 4-16
Clear Reference 4-16
Set Reference 4-16
4-13 :FETCh:VVM Subsystem 4-17
Data 4-18
4-14 FETCh:VVM:REFerence Subsystem 4-20
Reference Data 4-20

Chapter 5—Spectrum Analyzer Commands

5-1 Introduction 5-1
Spectrum Analyzer Commands 5-1
5-2 :ABORt Subsystem 5-2

5-3 :CALCulate Subsystem 5-3
Limit Alarm 5-3
Limit Fail 5-3
Lower Limit Fail State 5-4
Limit Pass/Fail 5-5
Upper Limit Fail State 5-6
Move Limit to Current Center Frequency 5-6
Create Limit Envelope 5-7
Limit Envelope Offset 5-7
Number of Limit Envelope Points 5-8
Limit Envelope Shape 5-8
Update Limit Envelope Frequency 5-9
Update Limit Envelope Amplitude 5-9
Limit Fail State 5-9
Lower Limit Alarm 5-10
Lower Limit Fail State 5-10
Number of Lower Limit Points 5-10
Lower Limit State 5-11
Limit Line Type 5-12
Limit Mirror 5-12
Add Limit Point 5-12
Delete Limit Point 5-13
Next Point Left 5-13
Next Point Right 5-13
Limit Point X Value 5-14
Limit Point Y Value 5-14
Number of Limit Points 5-15
Set Limit Line Upper or Lower 5-15
Upper Limit Alarm 5-16
Upper Limit Fail State 5-16
Number of Upper Limit Points 5-16
Upper Limit State 5-17
Move Limit 5-17
Add Vertical 5-18
Limit X Value 5-18
Limit Line Y Value 5-19
Set Default Limit 5-20
Limit State 5-20
Turn All Markers Off 5-21
Peak Threshold 5-21
Marker Data 5-21
Marker Table State 5-22

Marker Noise 5-22
Delta Marker X Value 5-23
Delta Marker Read Y Value 5-23
Delta Marker to Span 5-24
Delta Marker State 5-24
Marker Counter 5-25
Marker Fixed State 5-25
Marker (Maximum) Peak Search 5-26
Marker (Maximum) Peak Search Left 5-26
Marker (Maximum) Peak Search Next 5-26
Marker (Maximum) Peak Search Right 5-27
Marker Noise 5-27
Marker X Value 5-28
Marker Read Y Value 5-28
Marker Frequency to Center 5-29
Marker to Reference Level 5-29
Marker State 5-29

5-4 :CONFigure Subsystem 5-30
Configure Adjacent Channel Power Ratio 5-30
Configure Channel Power 5-30
Configure Field Strength 5-31
Configure Occupied Bandwidth 5-31
Measurement Mode 5-32

5-5 :DISPlay Subsystem 5-33
Display Grid 5-33
Auto Reference Level 5-33
Scale 5-33
Reference Level 5-34
Reference Level Offset 5-34

5-6 :FETCh Subsystem 5-35
Fetch Adjacent Channel Power Ratio 5-35
Fetch Channel Power 5-35
Fetch Channel Power Density 5-36
Fetch Channel Power/Density 5-36
Fetch Occupied Bandwidth Frequency 5-37
Fetch Occupied Bandwidth 5-37

5-7 :FORMat Subsystem 5-38
Numeric Data Format 5-38

5-8 :INITiate Subsystem 5-40
Continuous/Single Sweep 5-40
Trigger Sweep/Measurement 5-40
Save On Crossing Limit 5-41
Save On Sweep Complete 5-41
Save Then Stop 5-41
5-9 :MEASure Subsystem 5-42
Measure Adjacent Channel Power Ratio 5-42
Measure Channel Power 5-43
Measure Channel Power Density 5-44
Measure Channel Power/Density 5-44
Measure Occupied Bandwidth 5-45
5-10 :MMEMory Subsystem 5-46
Transfer Data 5-46
Delete Setup/Measurement 5-46
Recall Limit 5-47
Recall Setup 5-47
Recall Measurement 5-48
Save Limit 5-49
Save Setup 5-49
Save Measurement 5-50
5-11 :READ Subsystem 5-51
Read Adjacent Channel Power Ratio 5-51
Read Channel Power Density 5-52
Read Channel Power 5-52
Read Occupied Bandwidth 5-53
5-12 :TRACe Subsystem 5-54
Trace Copy 5-54
Trace Exchange 5-54
Trace Header Transfer 5-55
Trace Header Parameters 5-56
Trace Data Transfer 5-62
Trace View State 5-63
Trace Write State 5-64
Trace A Operation 5-65
Trace B Operation 5-66
Trace C Operation 5-67
5-13 :TRIGger Subsystem 5-68
Trigger Source 5-68
Video Trigger Level 5-68
Video Trigger Position 5-69

5-14 :UNIT Subsystem 5-70
Measurement Units 5-70
5-15 [:SENSe] Subsystem 5-71
ACPR Adjacent Channel Bandwidth 5-71
ACPR Main Channel Bandwidth 5-71
ACPR Channel Spacing 5-72
ACPR Measurement State 5-72
Number of Traces to Average 5-72
Trace Mode (Normal/Average/Max Hold/Min Hold). 5-73
Resolution Bandwidth 5-74
Resolution Bandwidth Coupling. 5-74
Resolution Bandwidth to Span Ratio 5-75
Video Bandwidth 5-75
Video Bandwidth Coupling 5-76
Video Bandwidth to Resolution Bandwidth Ratio 5-76
Channel Power Integration Bandwidth 5-77
Channel Power Measurement State 5-77
Input Impedance 5-78
Other Input Impedance Loss 5-78
Detection Type 5-79
Frequency Reference Status. 5-79
Center Frequency 5-80
Channel Selection 5-80
Signal Standard 5-81
Frequency Span 5-81
Frequency Span – Full 5-82
Frequency Span – Last 5-82
Start Frequency 5-82
Frequency Step 5-83
Stop Frequency 5-83
Field Strength Antenna 5-83
Field Strength Measurement State 5-84
Occupied Bandwidth Measurement Method 5-85
Occupied Bandwidth Percent of Power 5-85
Occupied Bandwidth Measurement State 5-86
Occupied Bandwidth dB Down 5-86
Input Attenuation 5-87
Input Attenuation Coupling 5-87
Preamp State. 5-88
Sweep Status 5-88
Minimum Sweep Time 5-88
Trace Count. 5-89

Chapter 6—AM/FM/PM Commands

6-1 :CALCulate Subsystem 6-1
6-2 :DISPlay Subsystem 6-5
6-3 :FORMat Subsystem 6-6
6-4 :INITiate Subsystem 6-8
6-5 :MMEMory Subsystem 6-9
6-6 :TRACe Subsystem 6-12
6-7 [:SENSe] Subsystem 6-13

Chapter 7—All Mode Commands

7-1 Introduction 7-1
7-2 All Mode Commands 7-1
7-3 :INSTrument Subsystem 7-2
Query Available Modes 7-2
Select Mode by Number 7-3
Select Mode by Name 7-4
7-4 :MMEMory Subsystem 7-5
Transfer Data 7-6
Delete Data/Location 7-6
Show Directory 7-7
7-5 :MMEMory:MSIS Subsystem 7-8
Storage Location 7-8
Copy From Current Location To Destination 7-8
Copy to Destination 7-9
7-6 :MMEMory:STORe Subsystem 7-10
Save Screen as JPEG 7-10
7-7 :SYSTem Subsystem 7-11
Query Installed Options 7-11
Preset 7-11
7-8 [:SENSe]:GPS Subsystem 7-12
GPS On/Off 7-12
GPS Reset 7-12
GPS Antenna Current 7-12
GPS Antenna Voltage 7-13
7-9 :FETCh:GPS Subsystem 7-14
Fetch GPS Fix Data 7-14

Appendix A—Example

A-1 Introduction A-1

A-2 C/C++ A-1
A-3 Visual Basic A-5

Appendix B—List of All SCPI Commands

Appendix C—SCPI Commands by Mode

Chapter 1 — General Information

1-1 About this Manual

This SCPI Programming Manual provides information for remote operation of the MS20xxB Vector Network Analyzer (MS2024B, MS2025B, MS2034B, and MS2035B) using commands sent from an external controller via Ethernet or USB connection. This Programming Manual includes the following:

- An overview of Ethernet and USB connection to the MS20xxB
- An overview of Standard Commands for Programmable Instruments (SCPI) command structure and conventions
- The IEEE common commands that are supported by the VNA Master
- A complete listing and description of all the SCPI commands that can be used to remotely control functions of the VNA Master. These commands are organized by instrument mode and are listed in Chapter 7, “All Mode Commands”.

This manual is intended to be used in conjunction with the VNA Master MS20xxB Vector Network Analyzer User Guide, Anritsu Part Number 10580-00301. Refer to that manual for general information about the VNA Master, including equipment setup and operating instructions.

1-2 Introduction

This chapter provides a general description of remote programming setup, USB and Ethernet interface connections, and cable requirements.

1-3 Remote Programming Setup and Interface

Remote programming and operation of the VNA Master is accessed via the USB or the Ethernet interface. The following paragraphs provide information about the interface connections, cable requirements, and setup for remote operation.

USB Interface Connection and Setup

Note	For proper detection, Master Software Tools must be installed on the PC prior to connecting to the MS20xxB using the USB port. Master Software Tools provides the installation tools to install the USB and VISA drivers.

The Universal Serial Bus (USB) architecture is a high-performance networking standard that is considered “plug and play” compatible. The USB driver software is automatically detected and configured by the operating system of the devices that are connected to the bus. The MS20xxB conforms to the USB 2.0 standard and is a USB “full-speed” device that supports data rates of up to 10 Mbps with the following restrictions:

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

To run the following example, you must have NI-VISA 2.5 or later installed on the controller PC, and you must select the VISA library (visa32.dll) as a reference in a Visual Basic project. For remote USB control, the controlling PC needs to have a version of VISA installed that supports USBTMC (USB Test and Measurement Class) devices.

1. Turn On power to the MS20xxB and controller PC and wait for the systems to power up completely.
2. Connect the USB cable mini-B connector to the MS20xxB.

3. Connect the USB cable A connector to the controller PC USB host port. The controller PC should indicate “New Hardware Found” if the combination of USB VID/PID/Serial Number has never been connected to this controller PC.

Figure 1-1. USB Found New Hardware Wizard

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

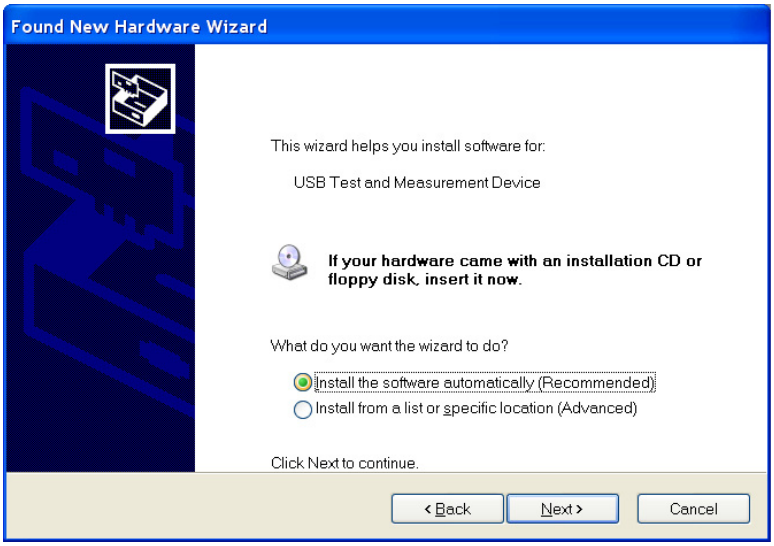


Figure 1-2. USB Found New Hardware Wizard

5. After the software is installed, close the Wizard by clicking Finish.

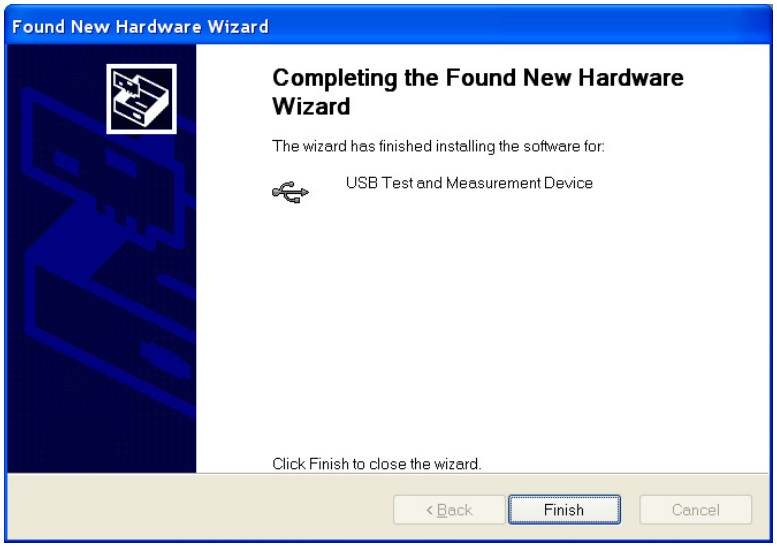


Figure 1-3. USB Found New Hardware Wizard

USB Interface, Type Mini-B

The USB 2.0 Mini-B device connector can be used to connect the MS20xxB directly to a PC. The first time that the MS20xxB is connected to a PC, the normal USB device detection is performed by the computer operating system. The driver is installed when Master Software Tools is installed (available from http://www.anritsu.com).

Ethernet Interface Connection and Setup

The VNA Master uses the Ethernet to communicate remotely with a controller. Most VNA Master functions (except Power On/Off) can be controlled via an Ethernet connection to a PC that is connected directly (with an Ethernet cross-over cable) or through a network. The instrument software supports the TCP/IP network protocol. (Ethernet connectivity was formerly supplied as Option 411. Revision 2 instruments have Ethernet connectivity built in.)

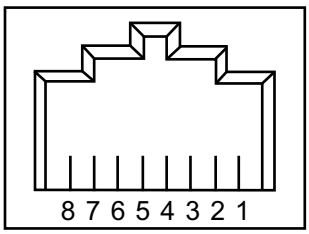
Ethernet networking uses a bus or star topology in which all of the interfacing devices are connected to a central cable called the bus, or are connected to a hub. Ethernet uses the CSMA/CD access method to handle simultaneous transmissions over the bus. CSMA/CD stands for *Carrier Sense Multiple Access/Collision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

The TCP/IP setup requires the following:

- IP Address: Every computer/electronic device in a TCP/IP network requires an IP address. An IP address has four numbers (each between 0 and 255) separated by periods. For example: 128.111.122.42 is a valid IP address.
- Subnet Mask: The subnet mask distinguishes the portion of the IP address that is the network ID from the portion that is the station ID. The subnet mask 255.255.0.0, when applied to the IP address given above, would identify the network ID as 128.111 and the station ID as 122.42. All stations in the same local area network should have the same network ID, but different station IDs.
- Default Gateway: A TCP/IP network can have a gateway to communicate beyond the LAN that is identified by the network ID. A gateway is a computer or electronic device that is connected to two different networks and can move TCP/IP data from one network to the other. A single LAN that is not connected to other LANs requires a default gateway setting of 0.0.0.0. If you have a gateway, then the default gateway would be set to the appropriate value of your gateway.
- Ethernet Address: An Ethernet address (also known as a MAC address) is a unique 48-bit value that identifies a network interface card to the rest of the network. Every network card has a unique ethernet address permanently stored into its memory.

Interface between the MS20xxB and other devices on the network is via a category-five (CAT-5) interface cable that is connected to a network. This cable uses 4 twisted pairs of insulated copper wires that are terminated at an RJ45 connector. CAT-5 cabling is capable of supporting frequencies up to 100 MHz and data transfer speeds up to 1 Gbps, which accommodates 1000Base-T, 100Base-T, and 10Base-T networks. CAT-5 cables are based on the EIA/TIA 568 Commercial Building Telecommunications Wiring Standard developed by the Electronics Industries Association. A pinout diagram is shown in Table 1-1.

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



Pin	Name	Description	Wire Color
1	TX+	Transmit data (> +3 volts)	White/Orange
2	TX–	Transmit data (< –3 volts)	Orange
3	RX+	Receive data (> +3 volts)	White/Green
4	–	Not used (common mode termination)	Blue
5	–	Not used (common mode termination)	White/Blue
6	RX–	Receive data (< –3 volts)	Green
7	–	Not used (common mode termination)	White/Brown
8	–	Not used (common mode termination)	Brown

Connectivity

TCP/IP connectivity requires setting up the parameters that are described at the beginning of this section. The following is a brief overview of how to set up a general LAN connection on the MS20xxB.

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

MS20xxB LAN Connections

The RJ-45 connector is used to connect the MS20xxB to a local area network (LAN). Integrated into this connector are two LEDs. The amber LED (Light Emitting Diode) indicates the speed of the LAN connection (ON for 10 Mb/s and OFF for 100 Mb/s), and the green LED flashes to show that LAN traffic is present. The instrument IP address is set by pressing the **Shift** key, then the **System** (8) key, then the System Options soft key, and then the Ethernet Config soft key. The instrument IP address can be set automatically by using DHCP, or can be set manually by entering the desired IP address, gateway address, and subnet mask.

Note	An active Ethernet cable must be connected to the MS20xxB before it is turned ON in order to enable the Ethernet port for DHCP or for a static IP address. Depending upon local conditions, the port may remain enabled when changing from DHCP to static IP address, when changing from static IP address to DHCP, or when temporarily disconnecting the Ethernet cable. If the port becomes disabled, ensure that an active Ethernet cable is attached to the MS20xxB, and then cycle the power OFF and back ON.

Dynamic Host Configuration Protocol (DHCP) is an Internet protocol that automates the process of setting IP addresses for devices that use TCP/IP, and is the most common method of configuring a device for network use. To determine if a network is set up for DHCP, connect the MS20xxB to the network and select DHCP protocol in the Ethernet Config menu.

Power cycle the MS20xxB. If the network is set up for DHCP, then the assigned IP address should be displayed briefly after the power-up sequence.

To display the IP address of the instrument, press the **Shift** key, then the **System** (8) key, then the System Options soft key, and then the Ethernet Config soft key.

1-4 Sending SCPI Commands

SCPI commands can be sent to the VNA Master though any Virtual Instrument Software Architecture (VISA) controller. VISA is a commonly-used API in the Test and Measurement industry for communicating with instruments from a PC. The physical connection between the PC and the VNA Master can be USB or Ethernet.

NI-VISA is the National Instruments implementation of the VISA I/O standard. Information and downloads are available at the following link:

http://www.ni.com/visa/

The following example describes the verification that a VISA controller can detect the VNA Master.

Note	The images shown and the instructions for your instrument and software may differ from the examples in this manual.

1. On the PC, run VISA Interactive Control and double click on the VNA Master.

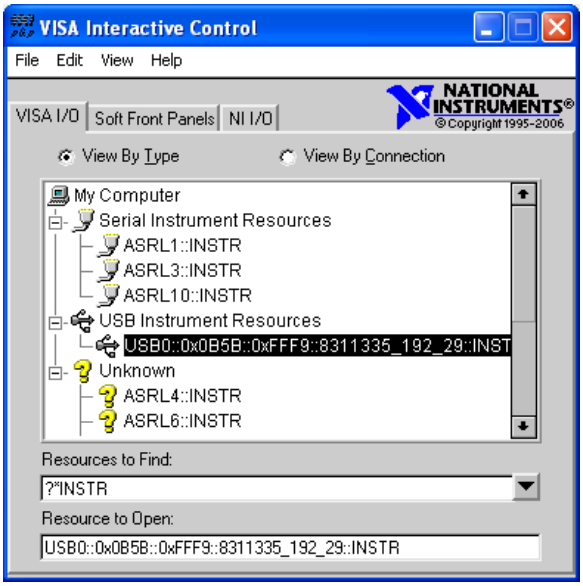


Figure 1-4. VISA Interactive Control

2. Select the viWrite tab and execute the default *IDN? write by clicking the Execute button.

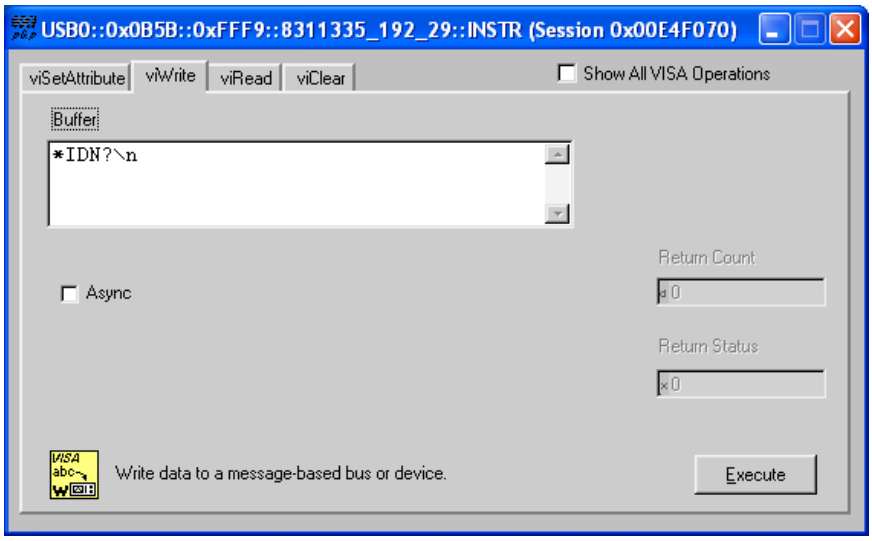


Figure 1-5. VISA Interactive Control viWrite Tab

3. Select the viRead tab and click the Execute button. If the PC is connected to the VNA Master, then the command returns the following information from the Buffer: manufacturer name ("Anritsu"), model number/options, serial number, and firmware package number.

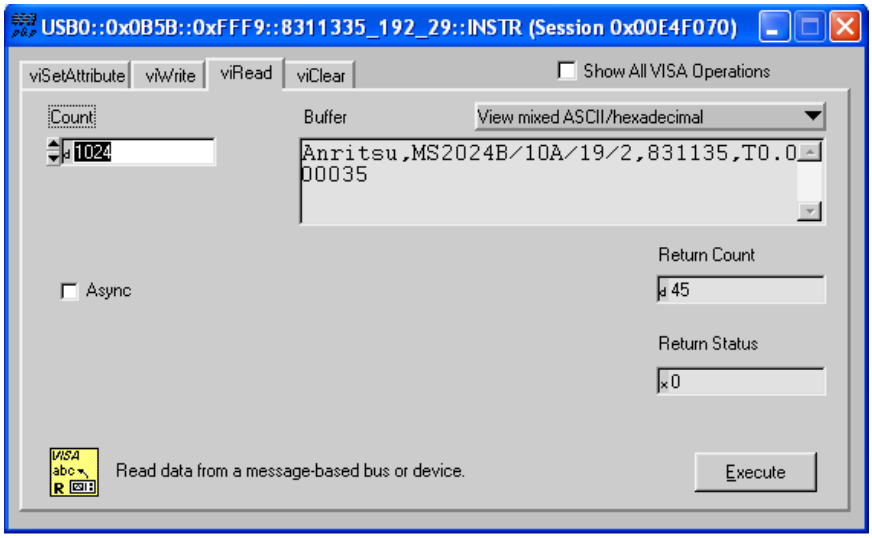


Figure 1-6. VISA Interactive Control viRead Tab

Chapter 2 — Programming with SCPI

2-1 Introduction

This chapter provides an introduction to SCPI programming that includes descriptions of the command types, hierarchical command structure, command subsystems, data parameters, and notational conventions.

2-2 Introduction to SCPI Programming

The Standard Commands for Programmable Instruments (SCPI) defines a set of standard programming commands for use by all SCPI-compatible instruments. SCPI is intended to give the user a consistent environment for program development. It does so by defining controller messages, instrument responses, and message formats for all SCPI-compatible instruments. SCPI commands are messages to the instrument to perform specific tasks. The MS20xxB command set includes:

- "SCPI Common Commands" on page 2-2
- "SCPI Required Commands" on page 2-3
- "SCPI Optional Commands" on page 2-3

Caution	Programs that receive SCPI commands may require support for Extended ASCII character codes in order to display some of the returned characters, such as Greek letter mu (μ). Some commands, for example, return the units of time in microseconds (μs). In this Anritsu programming manual, the Greek letter mu is represented by the English letter "u" to avoid typographic problems during publication.

Note	The MS20xxB follows the SCPI standard but is not fully compliant with that standard. The main reason that MS20xxB is not fully compliant is because it does not support all of the required SCPI commands, and because it uses some exceptions in the use of short form and long form command syntax. SCRE for SCREen and TYP for TYPE are two examples of the command short forms that are used in MS20xxB in order to be compatible with older products.

2-3 SCPI Common Commands

Some common commands are defined in the IEEE 488.2 standard and must be implemented by all SCPI compatible instruments. These commands are identified by the asterisk (*) at the beginning of the command keyword. These commands are defined to control instrument status registers, status reporting, synchronization, and other common functions. The common commands that are supported by the MS20xxB are shown below.

*IDN?

Title: Identification Query

Description: This command returns the following information in `<string>` format separated by commas: manufacturer name (“Anritsu”), model number/options, serial number, firmware package number. The model number and options are separated by a `"/"` and each option is separated by a `"/"`.

For example, the return string might appear as follows:

`"Anritsu,MS2028B/10/2,62011032,1.23"`

*RST

Title: Reset

Description: This command restores parameters in the current application as well as system settings to their factory default values.

System settings that are affected by this command are Ethernet configuration, language, volume, and brightness. Note that the unit will power cycle after this command is executed.

Front Panel Access: Shift-8 (System), System Options, Reset, Factory Defaults

See Also: `:SYSTem:PRESet`

Note	The best practice when starting any remote program is to reset the instrument to a known state. This is especially important when the instrument is being used in both remote and front-panel operation. Use the ***RST** or the **:SYSTem:PRESet** command to restore the instrument to the factory default settings. If using Ethernet to connect to the instrument, then note that the use of ***RST** resets the Ethernet configuration and causes the instrument to reboot, which may reset the instrument IP address. If resetting the Ethernet configuration is not desired, then you may choose to use **:SYSTem:PRESet**. Operating the instrument through the front panel and remotely at the same time could, under certain conditions, cause the instrument to hang up. To avoid conflicts, do not mix front-panel operation and remote operation.

2-4 SCPI Required Commands

The required SCPI commands that are supported by the MS20xxB are listed in the Table 2-1. These commands are described in Chapter 3 on page 3-1. Some `:SYSTem` commands function in all measurement modes, but the exact results of the commands may differ.

Table 2-1. SCPI Required Commands

`:STATus`
`:SYSTem`

2-5 SCPI Optional Commands

Table 2-2 lists the optional SCPI commands that comprise the majority of the command set that is described in this document. These commands control most of the programmable functions of the MS20xxB.

Table 2-2. SCPI Optional Commands

`:ABORt`	`:FETCh`	`:INSTrument`	`:SOURce`
`:CALCulate`	`:FORMat`	`:MEASure`	`:TRACe`
`:CALibration`	`:INITiate`	`:MMEMory`	`:UNIT`
`:DISPlay`	`:INPut`	`:SENSe`	`:[SENSe]`

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

- Chapter 3, “VNA Commands”
- Chapter 4, “Vector Voltmeter Commands”
- Chapter 5, “Spectrum Analyzer Commands”
- Chapter 7, “All Mode Commands”

2-6 Subsystem Commands

Subsystem commands control all instrument functions and some general purpose functions. All subsystem commands are identified by the colon that is used between keywords, as in `:INITiate:CONTinuous`.

The following information is provided for each subsystem command that is described in the following chapters:

- The command name ("Command Names" on page 2-4).
- The path from the subsystem root command ("Hierarchical Command Structure" on page 2-5).
- The query form of the command (if applicable) ("Query Commands" on page 2-7).
- The command title.
- A description of the purpose of the command.
- The data parameters that are used as arguments for the command (described in Section "Data Parameters" on page 2-8). This may include the parameter type and the available parameter choices.

Command Names

Typical SCPI commands consist of one or more keywords, parameters, and punctuation. SCPI command keywords can be a mixture of UPPERCASE and lowercase characters. Except for common commands, each keyword has a long form and a short form.

In this manual, the long form is presented with the short form portion in UPPERCASE and the remainder in lowercase. For example, the long form of the command keyword to control the instrument display is `:DISPlay`, and the short form is `:DISP`.

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

Some command keywords may have a numeric suffix to differentiate between multiple instrument features such as multiple trace options. For example; keywords `:TRACe[:DATA]{1|2|3}`, `:TRACe1`, or `:TRACe3`.

Note	In the previous paragraph, `:TRACe` is identical to `:TRACe1`. If a numeric suffix is not included in a command, then the first option is implied. Braces (curly brackets) `{ }` designate optional keyword parameters. Square brackets `[ ]` designate optional command keywords.

As with any programming language, the exact command keywords and command syntax must be used. The syntax of the individual commands is described in detail in the programming command chapters. Unrecognized versions of long form or short form commands, or improper syntax, will generate an error.

Long Format versus Short Format

Each keyword has a long format and a short format. The start frequency can be specified by `:SENSe:FREQuency:STARt` or `:SENS:FREQ:STAR`. The capital letters in the command specification indicate the short form of the command. A mixture of the entire short form elements with entire long form elements of each command is acceptable. For example, `:SENS:FREQuency:STAR` is an acceptable form of the command. However, `:SENS:FREQuen:STAR` is not an acceptable form of the command because `:FREQuen` is neither the short form nor the entire long form of the command element.

Hierarchical Command Structure

All SCPI commands, except the common commands, are organized in a hierarchical structure similar to the inverted tree file structure that is used in most computers. The SCPI standard refers to this structure as “the Command Tree.” The command keywords that correspond to the major instrument control functions are located at the top of the command tree. The root command keywords for the MS20xxB SCPI command set are shown in Figure 2-1.

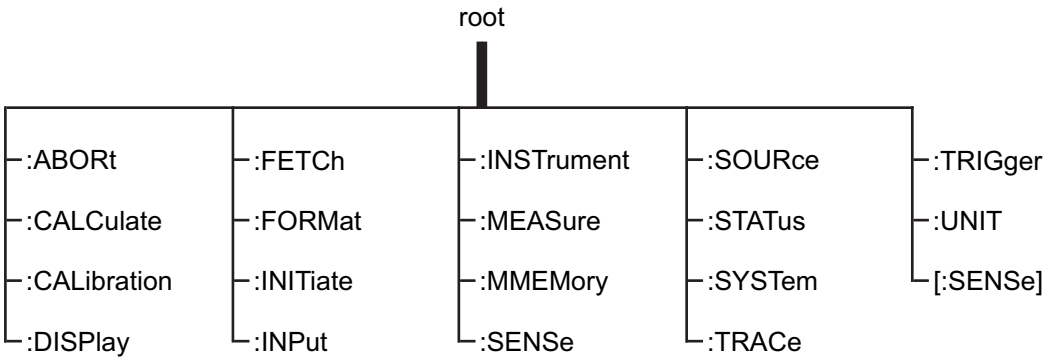


Figure 2-1. SCPI Command Tree

All MS20xxB SCPI commands, except the `:ABORt` command, have one or more subcommands (keywords) associated with them to further define the instrument function to be controlled. The subcommand keywords may also have one or more associated subcommands (keywords). Each subcommand level adds another layer to the command tree. The command keyword and its associated subcommand keywords form a portion of the command tree called a command subsystem. The `:DISPlay` command subsystem is shown in Figure 2-2.

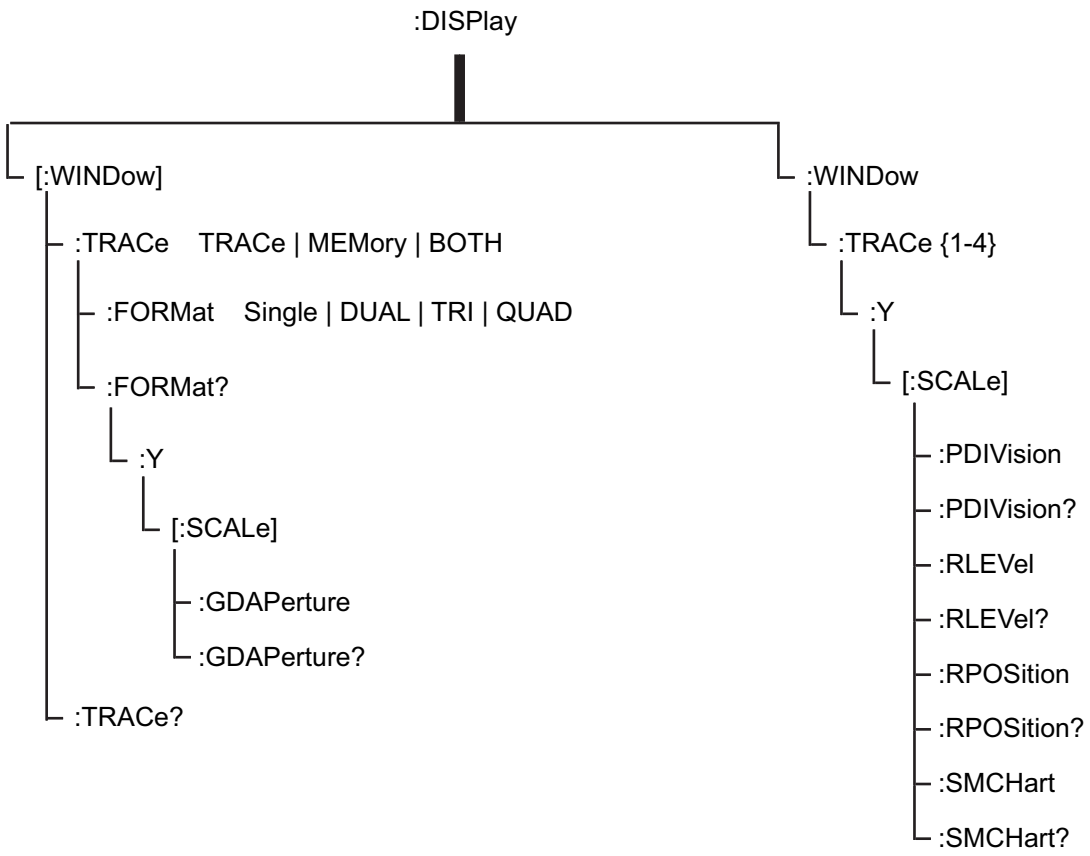


Figure 2-2. SCPI :DISPlay Subsystem

A colon (:) separates each subsystem. For example, the command `:DISPlay:WINDow:Trace MEMory` sets the window to display memory trace. Trace is part of the `:WINDow` subsystem, which is part of the `:DISPlay` subsystem. Y is also part of the `:DISPlay:WINDow:Trace{1-4}` subsystem.

Query Commands

All commands, unless specifically noted in the commands syntax descriptions, have a query form (refer also to Section 2-10 “Command and Query Notational Conventions” on page 2-12). As defined in IEEE-488.2, a query is a command with a question mark symbol appended (examples: `*IDN?` and `:TRACe[:DATA]? [1]|2|3|4)`. When a query form of a command is received, the current setting that is associated with the command is placed in the output buffer. Query commands usually return the short form of the parameter. Boolean values are returned as `1` or `0`, even when they can be set as `on` or `off`.

Note	When sending query commands immediately following an instrument setup command, a delay of up to two seconds may be required to allow the instrument sufficient time to complete the setup and receive the query.

Identifiers

Some or all of the following identifiers have been used throughout the optional command definitions. Descriptions are provided here. In most cases, units are specified with the individual command.

Table 2-3. Description of Command Indentifiers

Identifier	Description
`<amplitude>`	Amplitude value. Units specified with the command.
`<freq>`	Frequency. Units specified with the command.
`<integer>`	Integer value, no units. Range specified with the command.
`<number>`	Numeric value, integer or real.
`<percentage>`	Percentage value from 0 to 100. Units are always %.
`<rel ampl>`	Relative amplitude. Units are always dB.
`<x-parameter>`	Parameter value in the units of the x-axis. Units are specified with the command.
`<string>`	The string should be enclosed in either single quotes (‘ ’) or double quotes (“ ”).
`<filename>`	The name should be enclosed in either single quotes (‘ ’) or double quotes (“ ”). The need for an extension is documented with applicable commands.
`<voltage>`	Voltage. Units specified with the command.
`<current>`	Current. Units specified with the command.

Data Parameters

Data parameters, referred to simply as "parameters," are the quantitative values that are used as arguments for the command keywords. The parameter type that is associated with a particular SCPI command is determined by the type of information that is required to control the particular instrument function. For example, Boolean (`ON | OFF`) type parameters are used with commands that control switch functions.

Some command descriptions specify the type of data parameter that is to be used with each command. The most commonly used parameter types are numeric, extended numeric, discrete, and Boolean.

Numeric

Numeric parameters comprise integer numbers or any number in decimal or scientific notation, and may include polarity signs. This includes `<NR1>`, `<NR2>`, and `<NR3>` numeric data as defined in "Data Parameter Notations" on page 2-9. Parameters that accept all three `<NR>` formats are designated `<NRf>` throughout this document.

Extended Numeric

Extended numeric parameters include values such as `MAXimum` and `MINimum`.

Discrete

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

Boolean

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

Data Parameter Notations

The following syntax conventions are used for data parameter descriptions in this manual:

Table 2-4. Parameter Notations

<arg>	::=a generic command argument consisting of one or more of the other data types
<bNR1>	::=boolean values in <NR1> format; numeric 1 or 0
<boolean>	::=ON | OFF. Can also be represented as 1 or 0, where 1 means ON and 0 means OFF Boolean parameters are always returned as 1 or 0 in <NR1> format by query commands
<integer>	::=an unsigned integer without a decimal point (implied radix point)
<NR1>	::=a signed integer without a decimal point (implied radix point)
<NR2>	::=a signed number with an explicit radix point
<NR3>	::=a scaled explicit decimal point numeric value with an exponent (for example, floating point number)
<NRf>	::=<NR1>|<NR2>|<NR3>
<nv>	::=SCPI numeric value: <NRf>|MINimum|MAXimum|UP|DOWN|DEFault|NAN (Not A Number), |INFinity|NINFinity (Negative Infinity), or other types
<char>	::=<CHARACTER PROGRAM DATA> Examples: CW, FIXed, UP, and DOWN
<string>	::=<STRING PROGRAM DATA> ASCII characters surrounded by double quotes For example: “OFF”
<block>	::=IEEE-488.2 block data format
<NA>	::=Not Applicable

Unit Suffixes

Unit suffixes are not required for data parameters, provided the values are scaled for the global default units. The MS20xxB SCPI default units are: Hz (Hertz) for frequency-related parameters, s (seconds) for time-related parameters, and m (meters) for distance-related parameters.

If the command accepts a terminator, then the following are the available unit choices:

- `<freq>` accepts `GHZ` (Giga Hertz), `MHZ` or `MAHZ` (Mega Hertz), `KHZ` (Kilo Hertz), `HZ` (Hertz)
- `<time>` accepts `PS` (picosecond), `NS` (nanosecond), `US` (microsecond), `MS` (millisecond), `S` (Second)
- `<distance>` in meters accepts `MM` (millimeter), `M` (meter)
- `<distance>` in feet accepts `FT` (feet)

2-7 Notational Conventions

The SCPI interface standardizes command syntax and style to simplify the task of programming across a wide range of instrumentation. As with any programming language, the exact command keywords and command syntax must be used. Unrecognized commands or improper syntax will not function.

Table 2-5. Notational Conventions

:	A colon links command keywords together to form commands. The colon is not an actual part of the keyword, but is a signal to the SCPI interface parser. A colon must precede a root keyword immediately following a semicolon (see "Notational Examples" on page 2-11).
;	A semicolon separates commands if multiple commands are placed on a single program line.
[ ]	Square brackets enclose one or more optional keywords.
{ }	Braces enclose one or more keyword or command parameters that may be included one or more times.
|	A vertical bar indicates "or" and is used to separate alternative parameter options. Example: ON | OFF is the same as ON or OFF.
< >	Angle brackets enclose parameter descriptions.
::=	Means "is defined as". For example: <a>::=<b><c> indicates that <b><c> can replace <a>.
sp	Space, referred to as *white space*, must be used to separate keywords from their associated data parameters. It must not be used between keywords or inside keywords.
XXX	Indicates a root command name.

For further information about SCPI command syntax and style, refer to the Standard Commands for Programmable Instruments (SCPI) 1999.0 document.

2-8 Notational Examples

Table 2-6. Creating Valid Commands

Command Specification	Valid Forms

`[:SENSe]:FREQuency:STARt <freq>`	The following all produce the same result:
	`:SENSe:FREQuency:STARt 1 MHZ`
	`:SENS:FREQ:STAR 1 MHZ`
	`:sense:frequency:start 1000000`
	`:FREQ:STAR 1000 KHZ`

`:CALCulate:MARKer{1|2|3|4|5|6}:X <x-parameter>`	The first 2 commands set the location of marker 1. The third command sets the location of marker 2.
	`:CALC:MARK:X 1 GHZ`
	`:CALC:MARK1:X 1 GHZ`
	`:CALC:MARK2:X 2 GHZ`

`:UNIT:POWer DBM|DBV|DBMV|DBUV|V|W`	The following commands are identical:
	`:UNIT:POWer DBM`
	`:unit:pow dbm`

`:INITiate:CONTinuous OFF|ON|0|1`	The following commands are identical:
	`:INITiate:CONTinuous OFF`
	`:init:cont 0`

Command statements read from left to right and from top to bottom. In the command statement above, the `:FREQuency` keyword immediately follows the `:SENSe` keyword with no separating space. A space (*sp*) is used between the command string and its argument.

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

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

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

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

Command Terminators

The <new line> character (ASCII 10) in the last data byte of a command string is used as a command terminator. The use of a command terminator resets the command path to the root of the tree.

2-9 Formatting Conventions

This manual uses the following conventions in describing SCPI commands.

Table 2-7. Formatting Conventions

`:COMMands:LOOK:LIKE:THIS`	Commands are formatted to differentiate them from their description.
`:COMMand:QUERies:LOOK:LIKE:THIS?`	The query form of the command is followed by a “?”
Front panel key sequences use this formatting.	Front panel key presses are formatted to differentiate them from text descriptions. Key presses are separated by a comma (“,”).
`<identifier>`	Identifiers are enclosed in angular brackets, “< >”. They indicate that some type of data must be provided. Refer to Table 2-3 on page 2-7 for details on the types of identifiers.
`|`	The pipe (or vertical bar), “|” indicates that a choice must be made.
`[optional input]`	Optional input is enclosed in square brackets, “[ ]”. The “[ ]” are not part of the command.

2-10 Command and Query Notational Conventions

To distinguish the command types in the command descriptions, a question mark is included for a query.

2-11 Parameter Names

The parameters that are returned depend on the firmware version in the MS20xxB, and this document does not cover all possible parameter values that can be returned by the command.

Parameter names are dependent upon individual applications and are different for each application. They can be extracted via a Trace Preamble command.

The following tables list the parameter options for the `:TRACe:PREamble?` command in each supported measurement mode:

Vector Network Analyzer, refer to:

- Table 3-10, “Trace Header Parameters” on page 3-107.
- Table 3-11, “Trace Header Marker Parameters” on page 3-115
- Table 3-12, “Trace Header Limits Parameters” on page 3-116

Vector Voltmeter, refer to:

- Table 4-5, “Trace Header Parameters” on page 4-9.

Spectrum Analyzer, refer to:

- Table 5-2, “Available Parameters in Spectrum Analyzer Mode” on page 5-56.

Chapter 3 — VNA Commands

3-1 Introduction

This chapter describes commands for Vector Network Analyzer mode. Only the commands that are listed in this chapter and in Chapter 7, "All Mode Commands" can be used in Vector Network Analyzer mode. Using commands from other modes may produce unexpected results.

Note	Front Panel Access in VNA mode via the function hard keys may be listed as **Freq/Dist** or as **Freq**. The first function hard key is displayed with the label **Freq/Dist** when Option 501 is installed in the VNA Master.

Notational conventions are described in Section 2-10 "Command and Query Notational Conventions" on page 2-12.

VNA Commands

Table 3-1. VNA Commands Subsystems

Keyword	Parameter Data or Units
`:CALCulate{1-4}`	":CALCulate Subsystem" on page 3-2
`:DISPlay`	":Display Subsystem" on page 3-47
`:FORMat`	":Format Subsystem" on page 3-53
`:INITiate`	":INITiate Subsystem" on page 3-54
`:INPut`	":INPut Subsystem" on page 3-56
`:MMEMory`	":MMEMory Subsystem" on page 3-58
`[:SENSe]`	"[:SENSe] Subsystem" on page 3-63
`:SOURce`	":SOURce Subsystem" on page 3-100
`:STATus`	":STATus Subsystem" on page 3-102
`:SYSTem`	":SYSTem Subsystem" on page 3-103
`:TRACe`	":Trace Subsystem" on page 3-105

3-2 :CALCulate Subsystem

The commands in this subsystem process data that have been collected via the :CALCulate subsystem.

Table 3-2. :CALCulate Subsystem

Keyword	Parameter Data or Units
`:CALCulate{1-4}`	
`:FORMat`	Refer to “:CALCulate:FORMat Subsystem” on page 3-5
`:LIMit`	Refer to “:CALCulate:LIMit Subsystem” on page 3-6
`:MARKer`	Refer to “:CALCulate:MARKer Subsystem” on page 3-30
`:MATH`	Refer to “:CALCulate:MATH Subsystem” on page 3-42
`:SMOothing`	Refer to “:CALCulate:SMOothing Subsystem” on page 3-43
`:TRANsform`	Refer to “:CALCulate:TRANsform Subsystem” on page 3-44

Trace Data Transfer

:CALCulate<Tr>:DATA?

Description: Transfers the given trace data specified by <Tr> from the instrument to the controller. <Tr> is the trace number in the range 1 to 4. If no trace number is specified, then the <Tr> parameter defaults to trace number 1.

FDATa: Formatted (or Final) data. The returned data are based on the Graph Type that is associated with the trace. For graph types that use only one number per point (such as Log Mag, SWR, Phase, Real, Imaginary, Group Delay, Log Mag/2), the command returns one number per data point. For graph types that use two numbers per point (such as Smith Chart), the command returns two numbers per data point. Following is a list of the returned values for each Graph Type:

Graph Type	Returned Units
Log Magnitude	dB
Log Magnitude/2	dB
Phase	degree
SWR	unitless
Real	unitless
Imaginary	unitless
Group Delay	ns (nanosecond)
Smith Chart	R + jX ohm
Linear Polar	unitless, degree
Log Polar	dB, degree
Real Impedance	ohm
Imaginary Impedance	ohm

SDATa: Complex measurement data. The returned numbers (which are independent of the Graph Type that is associated with the trace) are the complex measurement data (Real and Imaginary) for each point of the trace. A 551 point trace therefore has a total of 1102 points that get transferred.

FMEM: Formatted (or Final) Memory data. Similar to FDATa, but for memory data.

SMEM: Complex memory data. Similar to SDATa, but for memory data.

Note that in order to get valid data when querying for memory data, you must first store a trace into memory using the command `CALC:MATH:MEMorize`. The format of the block data that is returned can be specified by the command `:FORMat:DATA`. The response begins with an ASCII header that specifies the number of data bytes. It appears in the format `#AX`, where `A` is the number of digits in `X`, and `X` is the number of bytes that follow the header. Each data point is separated by a comma delimiter.

Trace Data Transfer (continued)

Syntax: `:CALCulate<Tr>:DATA? FDATa|SDATa|FMEM|SMEM`
`:CALCulate{1-4}:DATA? FDATa|SDATa|FMEM|SMEM`

Cmd Parameter: **NA** (query only)

Query Parameter: `<char> FDATa|SDATa|FMEM|SMEM`

Query Response: `<block> (returns block data)`

Related Command: `:FORMat:DATA`

Front Panel Access: **NA**

3-3 :CALCulate:FORMat Subsystem

Commands in this subsystem define the display format for a measurement.

Graph Type

:CALCulate<Tr>:FORMat

Description: Defines the graph type for the given trace `<Tr>`. `<Tr>` is the trace number in the range 1 to 4. If no trace number is specified, then the `<Tr>` parameter defaults to trace number 1. `<Graph Type>` is the graph type to which the specified trace is set, and it must be one of the following values:

`LMAGnitude|SWR|PHASe|REAL|IMAGinary|GDELay|SMITh|LM/2|LINPolar|LOGPolar|RIMPedance|IIMPedance`

The query version of this command returns `"LMAG"` if the specified trace graph type is set to Log Mag, `"SWR"` if set to SWR, `"PHAS"` if set to Phase, `"REAL"` if set to Real, `"IMAG"` if set to Imaginary, `"GDEL"` if set to Group Delay, and `"SMIT"` is set to Smith chart, and `"LM/2"` if set to Log Mag/2 (cable loss), `"LINP"` if set to Linear Polar, `"LOGP"` if set to Log Polar, `"RIMP"` if set to Real Impedance, and `"IIMP"` if set to Imaginary Impedance.

Syntax:
```
:CALCulate<Tr>:FORMat <Graph Type>
:CALCulate{1-4}:FORMat <Graph Type>
:CALCulate<Tr>:FORMat?
```

Cmd Parameter:
```
<char> <Graph Type>
(LMAGnitude|SWR|PHASe|REAL|IMAGinary|GDELay|SMITh|LM/2|
LINPolar|LOGPolar|RIMPedance|IIMPedance)
```

Query Response:
```
<char> <Graph Type>
(LMAG|SWR|PHAS|REAL|IMAG|GDEL|SMIT|LM/2|LINP|
LOGP|RIMP|IIMP)
```

Default Value: Trace 1: SMIT

Trace 2: LMAG

Trace 3: LMAG

Trace 4: SMIT

Example: To set Trace 2 graph type to Log Magnitude

```
:CALC2:FORM LMAG
```

Front Panel Access: **Measure**, Graph Type

3-4 :CALCulate:LIMit Subsystem

This subsystem defines the limit lines and controls the limit check.

Limit Alarm

:CALCulate:LIMit:ALARm

Description: Enables/disables the active trace currently selected limit line alarm. Setting the value to `ON` or `1` turns on the active trace limit alarm. Setting the value to `OFF` or `0` turns off the active trace limit alarm. The query version of the command returns a `1` if the active trace currently selected limit line alarm is set to `ON` and returns `0` if set to `OFF`. Use `:CALCulate:LIMit:TYPE` to set the currently active limit line.

Syntax:
```
:CALCulate:LIMit:ALARm OFF|ON|0|1
:CALCulate:LIMit:ALARm?
```

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: OFF or 0 (query returns 0 for OFF)

Example: To turn off limit alarm

```
:CALCulate:LIMit:ALARm OFF
:CALCulate:LIMit:ALARm 0
```

To turn on limit alarm

```
:CALCulate:LIMit:ALARm ON
:CALC:LIM:ALAR 1
```

Related Command: `:CALCulate:LIMit:TYPE`

Front Panel Access: **Shift 6** (Limit), Limit Alarm

Number of Lower Limit Points

:CALCulate<Tr>:LIMit:LOWer:POINt?

Description: Returns the number of points currently in the lower limit line of the given trace `<Tr>`. `<Tr>` is the trace number in the range 1 to 4. If no trace number is specified, then the `<Tr>` parameter defaults to trace number 1.

Syntax: `:CALCulate<Tr>:LIMit:LOWer:POINt?`

Cmd Parameter: **NA** (query Only)

Query Response: `<NR1> <integer>`

Example: To query for the lower limit total point on trace #2:

```
:CALC2:LIM:LOW:POIN?
```

Front Panel Access: **NA**

Add Lower Limit Point

:CALCulate<Tr>:LIMit:LOWer:POINt:ADD

Description: Adds a new limit point to the lower limit line of the given trace `<Tr>`. `<Tr>` is the trace number in the range 1 to 4. If no trace number is specified, then the `<Tr>` parameter defaults to trace number 1.

Syntax: `:CALCulate<Tr>:LIMit:LOWer:POINt:ADD`

Cmd Parameter: **NA**

Query Response: **NA** (no query)

Example: To add a point to the lower limit line on trace 2:

`:CALC2:LIM:LOW:POIN:ADD`

Front Panel Access: **Shift 6** (Limit), Limit Edit, Add Point

Delete Lower Limit Point

:CALCulate<Tr>:LIMit:LOWer:POINt:DELete

Description: Deletes the lower limit point of the given trace `<Tr>`. After deletion, the point that is immediately to the left of the point that was deleted becomes the active point. `<Tr>` is the trace number in the range 1 to 4. If no trace number is specified, then the `<Tr>` parameter defaults to trace number 1. Note that deletion is valid only if more than 2 limit points are present.

Syntax: `:CALCulate<Tr>:LIMit:LOWer:POINt:DELete`

Cmd Parameter: **NA**

Query Response: **NA** (no query)

Example: To delete the trace 4 lower limit current active point:

`:CALCulate4:LIMit:LOWer:POINt:DELete`

Front Panel Access: **Shift 6** (Limit), Limit Edit, Delete Point

Lower Limit Next Point Left

:CALCulate<Tr>:LIMit:LOWer:POINt:LEFT

Description: Sets the limit point to the left of the lower limit active point of the given trace <Tr> as the new active point. <Tr> is the trace number in the range 1 to 4. If no trace number is specified, then the <Tr> parameter defaults to trace number 1.

Syntax: :CALCulate<Tr>:LIMit:LOWer:POINt:LEFT

Cmd Parameter: **NA**

Query Response: **NA** (no query)

Example: To make the lower limit point to the left of the current active point of trace 2 as the new active point:

:CALCulate2:LIMit:LOWer:POINt:LEFT

Front Panel Access: **Shift 6** (Limit), Limit Edit, Next Point Left

Lower Limit Next Point Right

:CALCulate<Tr>:LIMit:LOWer:POINt:RIGHt

Description: Sets the limit point to the right of the lower limit active point of the given trace <Tr> as the new active point. <Tr> is the trace number in the range 1 to 4. If no trace number is specified, then the <Tr> parameter defaults to trace number 1.

Syntax: :CALCulate<Tr>:LIMit:LOWer:POINt:RIGHt

Cmd Parameter: **NA**

Query Response: **NA** (no query)

Example: To make the lower limit point to the right of the current active point of trace 2 as the new active point:

:CALCulate2:LIMit:LOWer:POINt:RIGHt

Front Panel Access: **Shift 6** (Limit), Limit Edit, Next Point Right

Lower Limit Point X Value

:CALCulate<Tr>:LIMit:LOWer:POINt:X

Description: Sets the location of the lower limit point of the given trace <Tr> on the x-axis at the specified location. <Tr> is the trace number in the range 1 to 4. If no trace number is specified, then the <Tr> parameter defaults to trace number 1. <x-parameter> is defined in the current x-axis. Sending the set command changes the Move Limit on the front panel to Point if it is currently set to Limit, and sets the given trace as the active trace. The <x-parameter> given unit must correspond to the given trace domain type. If no unit is specified with the <x-parameter>, then the default unit is used. The query version of the command returns the location of the given trace active lower limit point on the x-axis followed by the unit. If an error occurs, such as limit not ON, then the query version of the command returns –400 error codes. Limit line must be ON for the command to be valid. Use :CALCulate:LIMit:TYPE to set the currently active limit line.

Syntax:
```
:CALCulate<Tr>:LIMit:LOWer:POINt:X <x-parameter>
:CALCulate<Tr>:LIMit:LOWer:POINt:X?
```

Cmd Parameter: <NRf> <x-parameter> (hertz, meters, feet)

Query Response: <NR3> <x-parameter> (hertz, meters, feet)

Default Units: Hz for Frequency domain,
Meters or Feet for distance domain.

Example: To set the trace 4 lower limit point to 5000 Hertz (trace 4 in frequency domain):

```
:CALCulate4:LIMit:LOWer:POINt:X 5000
```

OR to 500 MHz:

```
:CALCulate4:LIMit:LOWer:POINt:X 500 MHz
```

To set the trace 1 lower limit point to 5 Feet (trace 1 in distance domain with current distance unit in meter):

```
:CALCulate:LIMit:LOWer:POINt:X 5 FT
```

OR to 4 Meter

```
:CALCulate1:LIMit:LOWer:POINt:X 4 M
```

OR to 4 Meter

```
:CALCulate:LIMit:LOWer:POINt:X 4
```

Front Panel Access: **Shift 6** (Limit), Limit Edit, Limit X

Lower Limit Point Y Value

:CALCulate<Tr>:LIMit:LOWer:POINt:Y

Description: Sets the location of the lower limit point of the given trace <Tr> on the y-axis at the specified location. <Tr> is the trace number in the range 1 to 4. If no trace number is specified then default is trace number 1. Sending the set command changes the Move Limit on the front panel to Point if it is currently set to Limit, and sets the given trace as the active trace. The <y-parameter> is defined in the given trace current y-axis. If no unit is specified with the <y-parameter>, then the default unit is used. The query version of the command returns the location of the given trace lower limit point on the y-axis. If an error occurs, such as limit not ON, then the query version of the command returns –400 error codes. Limit line must be ON for the command to be valid. Use :CALCulate:LIMit:TYPE to set the currently active limit line.

Syntax:
```
:CALCulate<Tr>:LIMit:LOWer:POINt:Y <y-parameter>
:CALCulate<Tr>:LIMit:LOWer:POINt:Y?
```

Cmd Parameter: <NRf> <y-parameter> (depends on display type)

Query Response: <NR3> <y-parameter> (depends on display type)

Default Units: Current active trace y-axis unit

Related Command: :CALCulate:LIMit:TYPE

Front Panel Access: **Shift 6** (Limit), Limit Edit, Amplitude

Lower Limit State

:CALCulate<Tr>:LIMit:LOWer[:STATe]

Description: Turns the lower limit line of the given trace <Tr> ON or OFF. <Tr> is the trace number in the range 1 to 4. If no trace number is specified, then the <Tr> parameter defaults to trace number 1. <Tr> is the trace number in the range 1 to 4. If no trace number is specified, then the <Tr> parameter defaults to trace number 1. The query version of the command returns a 1 if the lower limit line of the given trace is ON, and returns a 0 if it is OFF.

Syntax:
```
:CALCulate<Tr>:LIMit:LOWer[:STATe] OFF|ON|0|1
:CALCulate<Tr>:LIMit:LOWer[:STATe]?
```

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: OFF or 0 (query returns 0 for OFF)

Example: To turn on lower limit of trace 1

```
:CALCulate:LIMit:LOWer ON
:CALCulate1:LIMit:LOWer 1
:CALCulate:LIMit:LOWer:STATe ON
```

To turn off upper limit of trace 4

```
:CALCulate4:LIMit:LOWer OFF
:CALCulate4:LIMit:LOWer 0
:CALC4:LIM:LOW:STAT 0
```

Related Command: `:CALCulate:LIMit[:STATe]`

Front Panel Access: **Shift 6** (Limit), Limit State

Lower Limit X Value

:CALCulate<Tr>:LIMit:LOWer:X

Description: Moves the lower limit of the given trace `<Tr>` on the x-axis to the given value. `<Tr>` is the trace number in the range 1 to 4. If no trace number is specified, then the `<Tr>` parameter defaults to trace number 1. `<x-parameter>` is defined in the given trace current x-axis. The unit given with the `<x-parameter>` must correspond to the given trace domain type. If no unit is specified with the `<x-parameter>`, then the default unit is used. The set version of the command changes the Move Limit on the front panel to Limit if it is currently set to Point, and sets the given trace as the active trace. The query version of the command returns the location of the given trace lower limit point on the x-axis followed by the unit. If an error occurs, such as limit not `ON`, then the query version of the command returns –400 error codes. Limit line must be ON for the command to be valid. Use `:CALCulate:LIMit:TYPE` to set the currently active limit line.

Syntax:
```
:CALCulate<Tr>:LIMit:LOWer:X <x-parameter>
:CALCulate<Tr>:LIMit:LOWer:X?
```

Cmd Parameter: `<NRf> <x-parameter> (hertz, meters, feet)`

Query Response: `<NR3> <x-parameter> (hertz, meters, feet)`

Default Units: Hz for Frequency domain,
Meters or Feet for distance domain.

Example: To move the trace 4 lower limit to 5000 Hertz (trace 4 in frequency domain)

```
:CALCulate4:LIMit:LOWer:X 5000
```

OR to 500 MHz:

```
:CALCulate4:LIMit:LOWer:X 500 MHz
```

To move the trace 1 lower limit to 5 Feet (trace 1 in distance domain with current distance unit in meter)

```
:CALCulate:LIMit:LOWer:X 5 FT
```

OR to 4 Meter

```
:CALCulate1:LIMit:LOWer:X 4 M
:CALCulate:LIMit:LOWer:X 4
```

Related Command:
```
:CALCulate:LIMit[:STATe]
:CALCulate<Tr>:LIMit:LOWer:Y
```

Front Panel Access: **Shift 6** (Limit), Limit Edit, Limit X

Lower Limit Y Value

:CALCulate<Tr>:LIMit:LOWer:Y

Description: Sets the location of the lower limit line of the given trace <Tr> on the y-axis at the given value. This moves the entire lower limit and moves the current active limit point by the given value. <Tr> is the trace number in the range 1 to 4. If no trace number is specified, then the <Tr> parameter defaults to trace number 1. The <y-parameter> is defined in the current y-axis. If no unit is specified with the <y-parameter>, then the default unit is used. The set version of the command changes the Move Limit on the front panel to Limit if it is currently set to Point, and sets the given trace as the active trace. The query version of the command returns the location of the active limit point on the y-axis. If an error occurs, such as limit not ON, then the query version of the command returns a –400 error codes. Limit line must be ON for the command to be valid.

Syntax:
```
:CALCulate<Tr>:LIMit:LOWer:Y <y-parameter>
:CALCulate<Tr>:LIMit:LOWer:Y?
```

Cmd Parameter: <NRf> <y-parameter> (depends on display type)

Query Response: <NR3> <y-parameter> (depends on display type)

Default Units: Current active trace y-axis unit

Related Command:
```
:CALCulate:LIMit[:STATe]
:CALCulate3:LIMit:LOWer:X
```

Front Panel Access: **Shift 6** (Limit), Limit Edit, Amplitude

Number of Limit Points

:CALCulate:LIMit:POINt?

Description: Returns the number of points currently in the selected limit line. Use :CALCulate:LIMit:TYPE to set the currently active limit line.

Syntax: :CALCulate:LIMit:POINt?

Cmd Parameter: **NA** (query only)

Query Response: <NR1> <integer>

Related Command: :CALCulate:LIMit:TYPE

Front Panel Access: **NA**

Add Limit Point

:CALCulate:LIMit:POINt:ADD

Description: Adds a new limit point to the currently active limit line. Use `:CALCulate:LIMit:TYPE` to set the currently active limit line.

Syntax: `:CALCulate:LIMit:POINt:ADD`

Cmd Parameter: **NA**

Query Response: **NA** (no query)

Related Command: `:CALCulate:LIMit:TYPE`
`:CALCulate:LIMit:POINt:DELete`

Front Panel Access: **Shift 6** (Limit), Limit Edit, Add Point

Delete Limit Point

:CALCulate:LIMit:POINt:DELete

Description: Deletes the active trace active limit point. After deletion, the point that is immediately to the left of the point that was deleted becomes the active point. Note that deletion is valid only if 2 or more limit points exist. Use `:CALCulate:LIMit:TYPE` to set the currently active limit line.

Syntax: `:CALCulate:LIMit:POINt:DELete`

Cmd Parameter: **NA**

Query Response: **NA** (no query)

Example: To delete the currently active limit point

`:CALCulate:LIMit:POINt:DELete`

Related Command: `:CALCulate:LIMit:POINt:ADD`

Front Panel Access: **Shift 6** (Limit), Limit Edit, Delete Point

Next Point Left

:CALCulate:LIMit:POINt:LEFT

Description: Sets the limit point immediately to the left of the active limit point as the active point. This makes it active for editing or deleting. Use `:CALCulate:LIMit:TYPE` to set the currently active limit line.

Syntax: `:CALCulate:LIMit:POINt:LEFT`

Cmd Parameter: **NA**

Query Response: **NA** (no query)

Example: To select the point to the left of the active point

`:CALCulate:LIMit:POINt:LEFT`

Related Command: `:CALCulate:LIMit:POINt:RIGHt`

Front Panel Access: **Shift 6** (Limit), Limit Edit, Next Point Left

Next Point Right

:CALCulate:LIMit:POINt:RIGHt

Description: Sets the limit point immediately to the right of the active limit point as the active point. This makes it active for editing or deleting. Use `:CALCulate:LIMit:TYPE` to set the currently active limit line.

Syntax: `:CALCulate:LIMit:POINt:RIGHt`

Cmd Parameter: **NA**

Query Response: **NA** (no query)

Example: To select the point to the right of the active point:

`:CALCulate:LIMit:POINt:RIGHt`

Related Command: `:CALCulate:LIMit:POINt:LEFT`

Front Panel Access: **Shift 6** (Limit), Limit Edit, Next Point Right

Limit Point X Value

:CALCulate:LIMit:POINt:X

Description: Sets the location of the active limit point on the x-axis at the specified location. Sending this command changes the Move Limit on the front panel to Point if it is currently set to Limit. The `<x-parameter>` must correspond to the current active trace domain type. If no unit is specified with the `<x-parameter>`, then the default unit is used. The query version of the command returns the location of the active limit point on the x-axis followed by the unit. If an error occurs, such as limit not `ON`, then the query version of the command returns –400 error codes. Limit line must be ON for the command to be valid. Use `:CALCulate:LIMit:TYPE` to set the currently active limit line.

Syntax:
```
:CALCulate:LIMit:POINt:X <x-parameter>
:CALCulate:LIMit:POINt:X?
```

Cmd Parameter: `<NRf> <x-parameter> (hertz, meters, feet)`

Query Response: `<NR3> <x-parameter> (hertz, meters, feet)`

Default Units: Hz for Frequency domain,
Meters or Feet for distance domain.

Example: To set the active limit point to 5000 Hertz (active trace in frequency domain):

```
:CALCulate:LIMit:POINt:X 5000
```

OR to 500 MHz:

```
:CALCulate:LIMit:POINt:X 500 MHz
```

To set the active limit point to 5 Feet (active trace in distance domain with current distance unit in meter):

```
:CALCulate:LIMit:POINt:X 5 FT
```

OR to 4 Meter

```
:CALCulate:LIMit:POINt:X 4 M
:CALCulate:LIMit:POINt:X 4
```

Related Command: `:CALCulate:LIMit:POINt:Y`

Front Panel Access: **Shift 6** (Limit), Limit Edit, Limit X

Limit Point Y Value

:CALCulate:LIMit:POINt:Y

Description: Sets the location of the active limit point on the y-axis at the specified location. Sending this command changes the Move Limit on the front panel to Point if it is currently set to Limit. The `<y-parameter>` is defined in the current y-axis. If no unit is specified with the `<y-parameter>`, then the default unit is used. The query version of the command returns the location of the active limit point on the y-axis. If an error occurs, such as limit not ON, the query version of the command returns an error code of –400. Limit line must be ON for the command to be valid. Use `:CALCulate:LIMit:TYPE` to set the currently active limit line.

Syntax:
```
:CALCulate:LIMit:POINt:Y <y-parameter>
:CALCulate:LIMit:POINt:Y?
```

Cmd Parameter: `<NRf> <y-parameter> (depends on display)`

Query Response: `<NR3> <y-parameter> (depends on display)`

Default Units: Current active trace y-axis unit

Related Command: `:CALCulate:LIMit:POINt:X`

Front Panel Access: **Shift 6** (Limit), Limit Edit, Amplitude

Limit State

:CALCulate:LIMit[:STATe]

Description: Turns the active trace currently selected limit line (upper or lower) ON or OFF. If the value is set to ON or 1, then the active trace selected limit line is turned ON. If the value is set to OFF or 0, then the active trace selected limit line is turned OFF. The query version of the command returns a 1 if the active trace selected limit line is ON and returns a 0 if it is OFF. Use `:CALCulate:LIMit:TYPe` to set the currently active limit line.

Syntax:
```
:CALCulate:LIMit[:STATe] OFF|ON|0|1
:CALCulate:LIMit[:STATe]?
```

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: OFF or 0 (query returns 0 for OFF)

Example: To turn on the currently selected limit line:

```
:CALCulate:LIMit ON
:CALCulate:LIMit:STATe ON
:CALCulate:LIMit:STATe 1
```

To turn off the currently selected limit line:

```
:CALCulate:LIMit OFF
:CALCulate:LIMit:STATe 0
:CALCulate:LIMit 0
```

Front Panel Access: **Shift 6** (Limit), Limit State

Limit Type

:CALCulate:LIMit:TYPE

Description: Sets the limit line segment type (upper or lower) to be edited. Set the value to 1 for Lower limit segment and to 0 for Upper limit line segment. The query version of the command returns a 1 if the lower limit line is currently active for editing and returns a 0 if the upper limit line is currently active for editing.

Syntax:
```
:CALCulate:LIMit:TYPE 0|1
:CALCulate:LIMit:TYPE?
```

Cmd Parameter: `<char> 0|1`

Query Response: `<char> 0|1`

Default Value: 0

Example: To set upper limit line active for editing:

```
:CALCulate:LIMit:TYPE 0
```

To set lower limit line active for editing:

```
:CALCulate:LIMit:TYPE 1
```

Front Panel Access: **Shift 6** (Limit), Limit

Number of Upper Limit Points

:CALCulate<Tr>:LIMit:UPPer:POINt?

Description: Query only. Returns the number of points currently in the upper limit line of the given trace `<Tr>`. `<Tr>` is the trace number in the range 1 to 4. If no trace number is specified, then the `<Tr>` parameter defaults to trace number 1.

Syntax: `:CALCulate<Tr>:LIMit:UPPer:POINt?`

Cmd Parameter: **NA** (query only)

Query Response: `<NR1> <integer>`

Example: To query for the upper limit line total point on trace #2:

```
:CALC2:LIM:UPP:POIN?
```

Front Panel Access: **NA**

Add Upper Limit Point

:CALCulate<Tr>:LIMit:UPPer:POINt:ADD

Description: Adds a new limit point to the upper limit line of the given trace `<Tr>`. `<Tr>` is the trace number in the range 1 to 4. If no trace number is specified, then the `<Tr>` parameter defaults to trace number 1.

Syntax: `:CALCulate<Tr>:LIMit:UPPer:POINt:ADD`

Parameter: **NA**

Query Syntax: **NA** (no query)

Query Response: **NA**

Example: To add a point to the upper limit line on trace 2:

```
:CALC2:LIM:UPP:POIN:ADD
```

Related Command: `:CALCulate<Tr>:LIMit:UPPer:POINt:DELete`

Front Panel Access: **Shift 6** (Limit), Limit Edit, Add Point

Delete Upper Limit Point

:CALCulate<Tr>:LIMit:UPPer:POINt:DELete

Description: Deletes the upper limit point of the given trace <Tr>. After deletion, the point that is immediately to the left of the point that was deleted becomes the active point. <Tr> is the trace number in the range 1 to 4. If no trace number is specified, then the <Tr> parameter defaults to trace number 1. Note that deletion is valid only if 2 or more limit points are active.

Syntax: `:CALCulate<Tr>:LIMit:UPPer:POINt:DELete`

Parameter: **NA**

Query Syntax: **NA** (no query)

Query Response: **NA**

Example: To delete trace 3 upper limit current active point:

```
:CALCulate3:LIMit:UPPer:POINt:DELete
```

Related Command: `:CALCulate<Tr>:LIMit:UPPer:POINt:ADD`

Front Panel Access: **Shift 6** (Limit), Limit Edit, Delete Point

Upper Limit Next Point Left

:CALCulate<Tr>:LIMit:UPPer:POINt:LEFT

Description: Sets the limit point to the left of the upper limit active point of the given trace <Tr> as the new active point. <Tr> is the trace number in the range 1 to 4. If no trace number is specified, then the <Tr> parameter defaults to trace number 1.

Syntax: `:CALCulate<Tr>:LIMit:UPPer:POINt:LEFT`

Parameter: **NA**

Query Syntax: **NA** (no query)

Query Response: **NA**

Example: To make the upper limit point to the left of the current active point of trace 2 as the new active point:

```
:CALCulate2:LIMit:UPPer:POINt:LEFT
:CALC2:LIM:UPP:POIN:LEFT
```

Related Command: `:CALCulate<Tr>:LIMit:UPPer:POINt:RIGHt`

Front Panel Access: **Shift 6** (Limit)), Limit Edit, Next Point Left

Upper Limit Next Point Right

:CALCulate<Tr>:LIMit:UPPer:POINt:RIGHt

Description: Sets the limit point to the right of the upper limit active point of the given trace <Tr> as the new active point. <Tr> is the trace number in the range 1 to 4. If no trace number is specified than defaults to trace number 1.

Syntax: :CALCulate<Tr>:LIMit:UPPer:POINt:RIGHt

Parameter: **NA**

Query Syntax: **NA** (no query)

Query Response: **NA**

Example: To make the upper limit point to the right of the current active point of trace 2 as the new active point:

```
:CALCulate2:LIMit:UPPer:POINt:RIGHt
:CALC2:LIM:UPP:POIN:RIGH
```

Related Command: :CALCulate<Tr>:LIMit:UPPer:POINt:LEFT

Front Panel Access: **Shift 6** (Limit), Limit Edit, Next Point Right

Upper Limit Point X Value

:CALCulate<Tr>:LIMit:UPPer:POINt:X

Description: Sets the location of the upper limit point of the given trace `<Tr>` on the x-axis at the specified location. `<Tr>` is the trace number in the range 1 to 4. If no trace number is specified than defaults to trace number 1. Sending the set command will change the Move Limit on the front panel to Point if it is currently set to Limit and set the given trace as the active trace. <x-parameter> is defined in the given trace current x-axis. The given unit must correspond to the given trace domain type. If no unit is specified with the <x-parameter> then the default unit will be used. The query version of the command returns the location of the given trace upper limit point on the x-axis followed by the unit. If an error occurs, such as limit not ON, the query version of the command returns an error code of –400. Limit line must be on for the command to be valid. Use `:CALCulate:LIMit:TYPe` to set the currently active limit line.

Syntax:
```
:CALCulate<Tr>:LIMit:UPPer:POINt:X <x-parameter>
:CALCulate<Tr>:LIMit:UPPer:POINt:X?
```

Cmd Parameter: `<NRf> <x-parameter> (hertz, meters, feet)`

Query Response: `<NR3> <x-parameter> (hertz, meters, feet)`

Default Units: Hz for Frequency domain,
Meters or Feet for distance domain.

Example: To set the trace 4 upper limit point to 5000 Hertz (trace 4 in frequency domain):

```
:CALCulate4:LIMit:UPPer:POINt:X 5000
```

OR to 500 MHz:

```
:CALCulate4:LIMit:UPPer:POINt:X 500 MHz
```

To set the trace 1 upper limit point to 5 Feet (trace 1 in distance domain with current distance unit in meter):

```
:CALCulate:LIMit:UPPer:POINt:X 5 FT
```

OR to 4 Meter

```
:CALCulate1:LIMit:UPPer:POINt:X 4 M
:CALCulate:LIMit:UPPer:POINt:X 4
```

Related Command:
```
:CALCulate:LIMit:TYPe
:CALCulate<Tr>:LIMit:UPPer:POINt:Y
```

Front Panel Access: **Shift 6** (Limit), Limit Edit, Limit X

Upper Limit Point Y Value

:CALCulate<Tr>:LIMit:UPPer:POINt:Y

Description: Sets the location of the upper limit point of the given trace <Tr> on the y-axis at the specified location. <Tr> is the trace number in the range 1 to 4. If no trace number is specified than defaults to trace number 1. Sending the set command will change the Move Limit on the front panel to Point if it is currently set to Limit and set the given trace as the active trace. The <y-parameter> is defined in the given trace current y-axis. If no unit is specified with the <y-parameter> then the default unit will be used. The query version of the command returns the location of the given trace upper limit point on the y-axis. If an error occurs, such as limit not ON, the query version of the command returns an error code of –400. Limit line must be on for the command to be valid. Use :CALCulate:LIMit:TYPe to set the currently active limit line.

Syntax:
```
:CALCulate<Tr>:LIMit:UPPer:POINt:Y <y-parameter>
:CALCulate<Tr>:LIMit:UPPer:POINt:Y?
```

Cmd Parameter: <NRf> <y-parameter> (depends on display type)

Query Response: <NR3> <y-parameter> (depends on display type)

Default Units: Current active trace y-axis unit

Related Command:
```
:CALCulate:LIMit:TYPe
:CALCulate<Tr>:LIMit:UPPer:POINt:X
```

Front Panel Access: **Shift 6** (Limit), Limit Edit, Amplitude

Upper Limit State

:CALCulate<Tr>:LIMit:UPPer[:STATe]

Description: Turns the upper limit line of the given trace `<Tr>` ON or OFF. `<Tr>` is the trace number in the range 1 to 4. If no trace number is specified than defaults to trace number 1. The query version of the command returns a 1 if the upper limit line of the given trace is ON and returns a Syntax

```
:CALCulate<Tr>:LIMit:UPPer[:STATe] OFF|ON|0|1
:CALCulate<Tr>:LIMit:UPPer[:STATe]?
```

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: OFF or 0 (query returns 0 for OFF)

Example: To turn on upper limit of trace 1:

```
:CALCulate:LIMit:UPPer ON
:CALCulate1:LIMit:UPPer 1
:CALCulate:LIMit:UPPer:STATe ON
```

To turn off upper limit of trace 4:

```
:CALCulate4:LIMit:UPPer OFF
:CALCulate4:LIMit:UPPer 0
:CALC4:LIM:UPP:STAT 0
```

Related Command: `:CALCulate:LIMit[:STATe]`

Front Panel Access: **Shift 6** (Limit), Limit State

Upper Limit X Value

:CALCulate<Tr>:LIMit:UPPer:X

Description: Moves the upper limit of the given trace `<Tr>` on the x-axis to the given value. `<Tr>` is the trace number in the range 1 to 4. If no trace number is specified, then the `<Tr>` parameter defaults to trace number 1. <x-parameter> is defined in the given trace current x-axis. The unit given with the <x-parameter> must correspond to the given trace domain type. If no unit is specified with the <x-parameter> then the default unit will be used. The set version of the command will change the Move Limit on the front panel to Limit if it is currently set to Point and set the given trace as the active trace. The query version of the command returns the location of the given trace upper limit point on the x-axis followed by the unit. If an error occurs, such as limit not ON, the query version of the command returns an error code of –400. Limit line must be on for the command to be valid. Use the command `:CALCulate:LIMit:TYPE` to set the currently active limit line.

Syntax:
```
:CALCulate<Tr>:LIMit:UPPer:X <x-parameter>
:CALCulate<Tr>:LIMit:UPPer:X?
```

Cmd Parameter: `<NRf> <x-parameter> (hertz, meters, feet)`

Query Response: `<NR3> <x-parameter> (hertz, meters, feet)`

Default Units: Hz for Frequency domain,
Meters or Feet for distance domain.

Example: To move the trace 4 upper limit to 5000 Hertz (trace 4 in frequency domain):

```
:CALCulate4:LIMit:UPPer:X 5000
```

OR to 500 MHz:

```
:CALCulate4:LIMit:UPPer:X 500 MHz
```

To move the trace 1 upper limit to 5 feet (trace 1 in distance domain with current distance unit in meter):

```
:CALCulate:LIMit:UPPer:X 5 FT
```

OR to 4 Meter

```
:CALCulate1:LIMit:UPPer:X 4 M
:CALCulate:LIMit:UPPer:X 4
```

Related Command:
```
:CALCulate:LIMit:TYPe
:CALCulate<Tr>:LIMit:UPPer:Y
```

Front Panel Access: **Shift 6** (Limit), Limit Edit, Limit X

Upper Limit Y Value

:CALCulate<Tr>:LIMit:UPPer:Y

Description: Sets the location of the upper limit line of the given trace `<Tr>` on the y-axis at the given value. This moves the entire upper limit and moves the current active limit point by the given value. `<Tr>` is the trace number in the range 1 to 4. If no trace number is specified than defaults to trace number 1. The <y-parameter> is defined in the current y-axis. If no unit is specified with the <y-parameter> then the default unit will be used. The set version of the command will change the Move Limit on the front panel to Limit if it is currently set to Point and set the given trace as the active trace. The query version of the command returns the location of the active limit point on the y-axis. If an error occurs, such as limit not ON, the query version of the command returns an error code of –400. Limit line must be on for the command to be valid. Use `:CALCulate:LIMit:TYPE` to set the currently active limit line.

Syntax:
```
:CALCulate<Tr>:LIMit:UPPer:Y <y-parameter>
:CALCulate<Tr>:LIMit:UPPer:Y?
```

Cmd Parameter: `<NRf> <y-parameter> (depends on display type)`

Query Response: `<NR3> <y-parameter> (depends on display type)`

Default Units: Current active trace y-axis unit

Related Command:
```
:CALCulate:LIMit:TYPe
:CALCulate<Tr>:LIMit:UPPer:X
```

Front Panel Access: **Shift 6** (Limit), Limit Edit, Amplitude

Limit X Value

:CALCulate:LIMit:X

Description: Sets the location of the active limit point on the x-axis at the specified location. This moves the entire limit and moves the active limit point to the given value. The <x-parameter> given unit must correspond to the current active trace domain type. If no unit is specified with the <x-parameter> then the default unit will be used. Sending the set command will change the Move Limit on the front panel to Limit if it is currently set to Point. The query version of the command returns the location of the active limit point on the x-axis followed by the unit. If an error occurs, such as limit not ON, the query version of the command returns an error code of –400. Limit line must be on for the command to be valid. Use `:CALCulate:LIMit:TYPE` to set the currently active limit line.

Syntax:
```
:CALCulate:LIMit:X <x-parameter>
:CALCulate:LIMit:X?
```

Cmd Parameter: `<NRf> <x-parameter> (hertz, meters, feet)`

Query Response: `<NR3> <x-parameter> (hertz, meters, feet)`

Default Units: Hz for Frequency domain,
Meters or Feet for distance domain.

Example: To move the active limit to 5000 Hertz (active trace in frequency domain):

```
:CALCulate:LIMit:X 5000
```

OR to 500 MHz:

```
:CALCulate:LIMit:X 500MHz
```

To move the active limit to 5 Feet (active trace in distance domain with current distance unit in feet):

```
:CALCulate:LIMit:X 5FT
```

OR to 4 Meter

```
:CALCulate:LIMit:X 4M
:CALCulate:LIMit:X 4
```

Related Command:
```
:CALCulate:LIMit:TYPe
:CALCulate:LIMit:Y
```

Front Panel Access: **Shift 6** (Limit), Limit Edit, Limit X

Limit Y Value

:CALCulate:LIMit:Y

Description: Sets the location of the active limit line on the y-axis at the given value. This moves the entire limit and moves the current active limit point by the given value. Sending this command will change the Move Limit on the front panel to Limit if it is currently set to Point. The <y-parameter> is defined in the current y-axis. If no unit is specified with the <y-parameter> then the default unit will be used. The query version of the command returns the location of the active limit point on the y-axis. If an error occurs, such as limit not ON, the query version of the command returns an error code of –400. Limit line must be on for the command to be valid. Use `:CALCulate:LIMit:TYPE` to set the currently active limit line.

Syntax:
```
:CALCulate:LIMit:Y <y-parameter>
:CALCulate:LIMit:Y?
```

Cmd Parameter: `<NRf> <y-parameter> (depends on display type)`

Query Response: `<NR3> <y-parameter> (depends on display type)`

Default Units: Current active trace y-axis unit

Related Command:
```
:CALCulate:LIMit:TYPe
:CALCulate:LIMit:X
```

Front Panel Access: **Shift 6** (Limit), Limit Edit, Amplitude

3-5 :CALCulate:MARKer Subsystem

This subsystem contains commands to manipulate data markers.

Turn All Markers Off

:CALCulate:MARKer:AOFF

Turns off all markers. This command will turn off all the markers and set Marker Type to off.

Marker Data

:CALCulate:MARKer:DATA?

Description: Reports the marker information. Each marker data is separated by a comma and data are returned similar to that when Readout Format is set to Table.

Syntax: `:CALCulate:MARKer:DATA?`

Cmd Parameter: **NA** (query only)

Query Response: `<char> (comma separated data)`

Front Panel Access: **NA**

Delta Marker Reference To

:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:DELTa:REFerence

Description: Sets the specified delta marker reference to the given reference marker specified by `<Mk>`. `<Mk>` is the reference marker number in the range of 1 to 12. The query version of the command returns the reference marker number to which the specified delta marker should be referenced. If the selected marker is not a delta marker, then –230 is returned. Note that the set version of this command will set the specified delta marker as the active marker. The given reference marker number must be currently set as a reference marker and the specified delta marker number must currently be set as delta marker. Also both markers (delta and reference) must be in the same domain type.

Syntax:

```
:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:DELTa:REF
erence <Mk>
:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:DELTa:REF
erence?
```

Cmd Parameter: `<char> <Mk>`

Query Response: `<char> <Mk>`

Default Value: –230 (The selected marker is a reference marker)

Example: Set Marker 1 as the reference marker of delta Marker 3:

```
:CALCulate:MARKer3:DELTa:REFerence 1
:CALC:MARK3:DELT:REF 1
```

Related Command: `:CALCulate:MARKer<Mk>:TYPE`, `:CALCulate:MARKer<Mk>:DOMain?`

Front Panel Access: **Marker**, Avail Ref Mkr

Delta Marker State

:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:DELTa[:STATe]

Description: Sets the specified marker as the active marker and turn it on or off. If the value is set to ON or 1, then the specified marker is turn on and set as a delta marker. If the value is set to OFF or 0, then the specified marker is turn off. The query version of the command returns a 1 if the specified marker is a delta marker, and returns a 0 if it is not a delta marker.

Syntax:

```
:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:DELTa[:ST
ATe]   OFF|ON|0|1
:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:DELTa[:ST
ATe]?
```

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: OFF

Example: To turn on marker #3 and set it as a delta marker:

```
:CALCulate:MARKer3:DELTa ON
:CALCulate:MARKer3:DELTa 1
:CALCulate:MARKer3:DELTa:STATe ON
:CALCulate:MARKer3:DELTa:STATe 1
```

To turn off delta marker #6:

```
:CALCulate:MARKer6:DELTa OFF
:CALCulate:MARKer6:DELTa:STATe OFF
:CALCulate:MARKer6:DELTa:STATe 0
```

Related Command: `:CALCulate:MARKer:DELTa:REFerence`

Front Panel Access: **Marker**, Maker Type

Marker Readout Format

:CALCulate:MARKer:DISPlay:FORMat

Description: Sets the display readout format for markers. The query version of the command returns “NONE” if the display readout format is set to None, “SCRE” if Screen, “TABL” if Table, and “TRAC” if Trace.

Syntax:
```
:CALCulate:MARKer:DISPlay:FORMat
NONE|SCREen|TABLe|TRACe
:CALCulate:MARKer:DISPlay:FORMat?
```

Cmd Parameter: `<char> NONE|SCREen|TABLe|TRACe`

Query Response: `<char> NONE|SCRE|TABL|TRAC`

Default Value: NONE

Example: To set marker readout format to Table:

```
:CALCulate:MARKer:DISPlay:FORMat TABLe
:CALCulate:MARKer:DISPlay:FORMat TABL
:CALC:MARK:DISP:FORM TABL
```

Front Panel Access: **Marker**, Readout Format

Marker Domain Type

:CALCulate:MARKer<Mk>:DOMain?

Description: Query the specified marker `<Mk>` domain type. `<Mk>` is the marker number in the range of 1 to 12. If no marker number is specified, then the marker number (the `<Mk>` value) defaults to 1. This command returns “FREQ” if the specified marker domain is frequency and “DIST” if distance.

Syntax: `:CALCulate:MARKer<Mk>:DOMain?`

Cmd Parameter: **NA** (query only)

Query Response: `<char> FREQ|DIST`

Front Panel Access: **NA**

Marker Readout Style

:CALCulate:MARKer<Mk>:FORMat

Description: Sets the specified marker <Mk> readout style. <Mk> is the marker number in the range of 1 to 12. If no marker number is specified, then the marker number (the <Mk> value) defaults to 1. <Style> is the marker readout style and must be one of the following values:

GRAPh|LMAGnitude|LOGPhase|PHASe|RLIMaginary|SWR|IMPedance||ADMittance|NIMPedance|NADMittance|PIMPedance|GDELay|LM/2|LINMagnitude|LINPhase

The query version of the command returns "GRAP" if the specified marker readout style is set to Graph Type, "LMAG" if the specified marker readout style is set to Log Magnitude, "LOGP" if Log Mag and Phase, "PHAS" if Phase, "RLIM" if Real and Imaginary, "SWR" if standing wave ratio, "IMP" for impedance, "ADM" for admittance, "NIMP" for normalized impedance, "NADM" for normalized admittance, "PIMP" for polar impedance, "GDEL" if group delay, "LM/2" for log mag/2 (cable loss), "LINM" for Linear Magnitude (Lin Mag), and "LINP" for Linear Magnitude and Phase. Note that the set version of this command will set the specified marker as the active marker.

Syntax:
```
:CALCulate:MARKer<Mk>:FORMat <Style>
:CALCulate:MARKer<Mk>:FORMat?
```

Cmd Parameter: <char> <Style> (GRAPh|LMAGnitude|LOGPhase|and so forth)

Query Response: <char> <Style> (GRAP|LMAG|LOGP|and so forth)

Default Value: GRAP

Example: To set marker #3 readout style to Log Mag:

```
:CALCulate:MARKer3:FORMat LMAG
:CALCulate:MARKer3:FORMat LMAGnitude
:CALC:MARK3:FORM LMAG
```

Related Command:
```
:CALCulate:MARKer<Mk>:Y?,
:CALCulate:MARKer:DATA?
```

Front Panel Access: **Marker**, Readout Style

Marker (Maximum) Peak Search

:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:MAXimum

Description: Puts the specified marker at the maximum value in the trace. Note that this turns on the selected marker (if it is not already on) and sets the selected marker as the active marker.

Syntax: `:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:MAXimum`

Cmd Parameter: **NA**

Query Response: **NA** (no query)

Related Command: `:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:MINimum`

Front Panel Access: **Marker**, Marker Search, Peak Search

Marker (Minimum) Valley Search

:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:MINimum

Description: Puts the specified marker at the minimum value in the trace. Note that this turns on the selected marker (if it is not already on) and set the selected marker as the active marker.

Syntax: `:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:MINimum`

Cmd Parameter: **NA**

Query Response: **NA** (no query)

Related Command: `:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:MAXimum`

Front Panel Access: **Marker**, Marker Search, Valley Search

Reference Marker State

:CALCulate:MARKer<Mk>:REFerence[:STATe]

Description: Sets the specified marker <Mk> as the active marker and turn it on or off. If the value is set to ON or 1, the specified marker is turn on and set as a reference marker. If the value is set to OFF or 0, the specified marker is turn off. The query version of the command returns a 1 if the specified marker is ON and is a reference marker, and returns a 0 if not. <Mk> is the marker number in the range of 1 to 12. If no marker number is specified, then the marker number (the <Mk> value) defaults to 1.

Syntax:
```
:CALCulate:MARKer<Mk>:REFerence[:STATe] OFF|ON|0|1
:CALCulate:MARKer<Mk>:REFerence[:STATe]?
```

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: OFF

Example: To turn on marker #3 and set it as reference marker:

```
:CALCulate:MARKer3:REFerence ON
:CALCulate:MARKer3:REFerence 1
:CALCulate:MARKer3:REFerence:STATe ON
:CALCulate:MARKer3:REFerence:STATe 1
```

To turn off marker #6:

```
:CALCulate:MARKer6:REFerence OFF
:CALCulate:MARKer6:REFerence:STATe OFF
:CALCulate:MARKer6:REFerence:STATe 0
```

Front Panel Access: **Marker**, Marker Type

Marker On Trace

:CALCulate:MARKer<Mk>:SOURce

Description: Sets the specified marker <Mk> to the given trace <Tr>. <Mk> is the marker number in the range of 1 to 12. If no marker number is specified, then the marker number (the <Mk> value) defaults to 1. <Tr> is the trace and must be one of the following 9 values:

TR1|TR2|TR3|TR4|MEM1|MEM2|MEM3|MEM4|ALL

The query version of the command returns "TR1" if the specified marker is on trace 1, "TR2" if on trace 2, "TR3" if on trace 3, "TR4" if on trace 4, "MEM1" if on trace 1 memory, "MEM2" if on trace 2 memory, "MEM3" if on trace 3 memory, "MEM4" if on trace 4 memory, and "ALL" if the specified marker is on all 4 traces. Note that the set version of this command will set the specified marker as the active marker.

If an error occurs, such as "Marker not ON", the query version of the command returns an error code of -400.

Syntax:
```
:CALCulate:MARKer<Mk>:SOURce <Tr>
:CALCulate:MARKer<Mk>:SOURce?
```

Cmd Parameter: <char> <Tr>

Query Response: <char> <Tr>

Front Panel Access: **Marker**, Marker on Trace

Marker Type

:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:TYPE

Description: Sets the specified marker to the given marker type and make it the active marker. If set to REF then the specified marker is turn on and is set as reference marker. If set to DELT then the specified marker is turn on and is set as a delta marker. If set to OFF then the specified marker is turn off. The query version of this command returns the string “REF” if the specified marker is set as reference marker, “DELT” if set as delta marker, or “OFF” if the specified marker is currently set to off.

Syntax:
```
:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:TYPE
  REFerence|DELTa|OFF
:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:TYPE?
```

Cmd Parameter: `<char> REFerence|DELTa|OFF`

Query Response: `<char> REF|DELT|OFF`

Default Value: OFF

Example: To set marker #1 as the reference marker and turn it on:

```
:CALCulate:MARKer1:TYPE REFerence
:CALCulate:MARKer:TYPE REF
```

Front Panel Access: **Marker**, Marker Type

Marker X Value

:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:X

Description: Sets the location of the marker on the x-axis at the specified location. <x-parameter> is defined in the current x-axis units. The set command will set the specified marker as the active marker. The <x-parameter> given unit must correspond to the specified marker domain type. If no unit is specified with the <x-parameter> then the default unit will be used.

The query version of the command returns the location of the marker on the x-axis followed by the unit. If the Start and Stop values of the domain are the same, then the query returns the X value along with the marker point number within the brackets. The marker point number is determined with the following formula:

$$\text{Marker Point Number} = \frac{\text{No. of Points}}{2} + 1$$

For example, if 201 points are used in the measurement display, then the query returns: `<x-value> (101)`. When the results of division include a fraction, as in the current example, the result is rounded DOWN. `201/2 = 100.5` Rounding down to 100 before adding 1 yields the 101 that is returned by the query.

If an error occurs, such as marker not ON, the query version of the command returns an error code of –400. Note that the marker is snapped to the data point closest to the specified value. The selected marker must be ON for the command to be valid.

Syntax:
```
:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:X
<x-parameter>
:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:X?
```

Cmd Parameter: `<NRf> <x-parameter> (hertz, meters, feet)`

Query Response: `<NR3> <x-parameter> (hertz, meters, feet)`

Default Units: Hz for frequency domain,
Meters or Feet for distance domain.

Example: To set reference marker #2 (frequency domain) to 5000 hertz on the x-axis:

```
:CALCulate:MARKer2:X 5000
:CALCulate:MARKer2:X 5000Hz
```

To set reference marker #1 to 1.5 GHz on the x-axis:

```
:CALCulate:MARKer1:X 1.5GHz
:CALCulate:MARKer1:X 1.5GHz
```

Related Command:
```
:CALCulate:MARKer#:DOMain?
:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:Y?
```

Front Panel Access: **Marker**, [Marker 1/2/3/4/5/6/7/8/9/10/11/12]

Marker Read Y Value

:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:Y?

Description: Reads the current Y value for the specified marker. The units are in the y-axis unit. The command returns the marker readout style followed by the Y value and unit. If an error occurs, such as marker not ON, the command returns an error code of –400. The selected marker must be ON for the command to be valid.

Table 3-3. Returned Readout Style

Returned Value Symbols	Graph Type
R&I: (real, imaginary)	Real and Imaginary
SWR: magnitude	SWR
LM: magnitude dB	Log Mag
LNM: magnitude dB	Lin Mag
LMP: (magnitude dB, phase deg)	Log Mag & Phase
LNMP: (magnitude dB, phase deg)	Lin Mag and Phase
PH: phase deg	Phase
Z: (real impedance ohm, imaginary impedance ohm)	Impedance
PZ: (magnitude impedance ohm, phase impedance deg)	Polar Impedance
NZ: (real normalized impedance, imaginary normalized impedance)	Normalized Impedance
Y: (real admittance S, imaginary admittance S)	Admittance
NY: (real normalized admittance, imaginary normalized admittance)	Normalized Admittance
GD: group delay unit	Group Delay
LM/2: magnitude dB	LogMag/2

Marker Read Y Value (continued)

Syntax: `:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:Y?`

Cmd Parameter: **NA** (query only)

Query Response: `<NR3> (depends on display type)`

Default Units: Current y-axis unit

Related Command:
```
:CALCulate:MARKer#:DOMain?
:CALCulate:MARKer<Mk>:FORMat <Style>
:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:X?
```

Front Panel Access: **NA**

3-6 :CALCulate:MATH Subsystem

This subsystem contains functions for controlling math operations on the currently selected measurement and memory.Trace Math Function

:CALCulate:MATH:FUNCtion

Description: Sets the math operations on the currently active trace and the trace stored in memory. Note that a trace MUST be stored in Memory. Setting the FUNCtion to NORMal is equivalent of setting the Trace Math to “None” on the front panel. Setting the FUNCtion to ADD is equivalent of setting the Trace Math to “Trace Plus Memory” on the front panel. Setting the FUNCtion to SUBTract is equivalent to setting the Trace Math to “Trace Minus Memory” on the front panel. Setting the FUNCtion to MULTiply is equivalent to setting the Trace Math to “Trace Multiply Memory” on the front panel. Setting the FUNCtion to DIVide is equivalent to setting the Trace Math to “Trace Divide Memory” on the front panel. The query version of the command returns the string “NORM” for no trace math, “ADD” for trace plus memory, “SUBT” for trace minus memory, “MULT” for trace multiply memory, and “DIV” for trace divide memory.

Syntax: `:CALCulate:MATH:FUNCtion NORMal|ADD|SUBTract|MULTiply|DIVide`
`:CALCulate:MATH:FUNCtion?`

Cmd Parameter: `<char> NORMal|ADD|SUBTract|MULTiply|DIVide`

Query Response: `<char> NORM|ADD|SUBT|MULT|DIV`

Default Value: NORM

Related Command: `:CALCulate:MATH:MEMorize`

Front Panel Access: **Shift 5** (Trace), Trace Math

Trace To Memory

:CALCulate:MATH:MEMorize

Description: Copies the current measurement trace into memory.

Syntax: `:CALCulate:MATH:MEMorize`

Cmd Parameter: **NA**

Query Response: **NA** (no query)

Front Panel Access: **Shift 5** (Trace), Save Trace to Memory

3-7 :CALCulate:SMOothing Subsystem

This subsystem contains functions for trace smoothing.

Smoothing

:CALCulate<Tr>:SMOothing:APERture

Description: Sets the smoothing percentage for the given trace `<Tr>`. The query form of the command returns the current smoothing percentage. `<Tr>` is the trace number in the range 1 to 4. If no trace number is specified, then the `<Tr>` parameter defaults to trace number 1. Note that setting the smoothing also sets the given trace as the active trace if it is not already active.

Syntax:
```
:CALCulate<Tr>:SMOothing:APERture <integer>
:CALCulate<Tr>:SMOothing:APERture?
```

Cmd Parameter: `<NR1> <integer>`

Query Response: `<NR1> <integer>`

Default Value: 0

Range: 0 to 20

Front Panel Access: **Shift 4** (Measure), Smoothing %

3-8 :CALCulate:TRANsform Subsystem

Front panel soft keys that are related to distance measurements, such as the Additional Dist Setup soft key, appear in menus only when the Setup Domain is set up for distance.

Maximum Distance

:CALCulate:TRANsform:DISTance:MAXimum?

Description: This command returns the maximum distance in millimeters if the current distance unit is set to meter, and otherwise returns the maximum distance in feet. This value is set based on the number of data points, the propagation velocity, and the start and stop frequency.

Syntax: `:CALCulate:TRANsform:DISTance:MAXimum?`

Cmd Parameter: **NA** (query only)

Query Response: `<NR3> (millimeters or feet)`

Default Value: NA – depends upon frequency settings

Default Unit: millimeters (mm)

Range: –3000.0 m to +3000.0 m

Front Panel Access: **Freq/Dist**, Additional Dist Setup, Distance Info

Distance Resolution

:CALCulate:TRANsform:DISTance:RESolution?

Description: This command returns the distance resolution in millimeters if the current distance unit is set to meter, and otherwise returns the resolution in feet. This value is set based on the propagation velocity and the start and stop frequencies.

Syntax: `:CALCulate:TRANsform:DISTance:RESolution?`

Cmd Parameter: **NA** (query only)

Query Response: `<NR3> (millimeters or feet)`

Default Value: 4.997 m for 6 GHz instruments
7.496 m for 4 GHz instruments

Default Unit: millimeters (mm)

Range: –3000.0 m to +3000.0 m

Front Panel Access: **Freq/Dist**, Additional Dist Setup, Distance Info

Start Distance

:CALCulate:TRANsform:DISTance:STARt

Description: Sets the start distance for DTF measurements. The query version of this command returns the start distance in millimeters if the current distance unit is set to meter, and otherwise returns the start distance in feet.

Syntax:
```
:CALCulate:TRANsform:DISTance:STARt
:CALCulate:TRANsform:DISTance:STARt?
```

Cmd Parameter: `<NRf> (meters or feet)`

Query Response: `<NR3> (millimeters or feet)`

Default Value: 0.0 mm

Default Unit: Meters (m) when setting, Millimeters (mm) for query

Range: –3000.0 m to +3000.0 m

Example: To set the start distance to 5 meters:

```
:CALC:TRAN:DIST:STAR 5
```

To set the start distance to 6 millimeters:

```
:CALCulate:TRANsform:DISTance:STARt 6mm
```

Front Panel Access: **Freq/Dist**, Start Dist

Stop Distance

:CALCulate:TRANsform:DISTance:STOP

Description: Sets the stop distance for DTF measurements. The query version of this command returns the stop distance in millimeters if the current distance unit is set to meter, and otherwise returns the stop distance in feet.

Syntax:
```
:CALCulate:TRANsform:DISTance:STOP
:CALCulate:TRANsform:DISTance:STOP?
```

Cmd Parameter: `<NRf> (meters or feet)`

Query Response: `<NR3> (millimeters or feet)`

Default Value: 6850 mm

Default Unit: Meters (m) when setting, Millimeters (mm) for query

Range: –3000.0 m to +3000.0 m

Front Panel Access: **Freq/Dist**, Stop Dist

Distance Units

:CALCulate:TRANsform:DISTance:UNIT

Description: Sets the units to be used for DTF measurements. The query version of this command returns the string "METER" if the current distance unit is set to meter, and otherwise returns the string "FEET".

Syntax:
```
:CALCulate:TRANsform:DISTance:UNIT METers|FEET
:CALCulate:TRANsform:DISTance:UNIT?
```

Cmd Parameter: `<char> METers|FEET`

Query Response: `<char> METER|FEET`

Default Value: METers when setting, METER for query

Example: To set the distance unit to Meter:

```
:CALCulate:TRANsform:DISTance:UNIT METers
:CALC:TRAN:DIST:UNIT MET
```

To set the distance unit to Feet:

```
:CALC:TRAN:DIST:UNIT FEET
:CALCulate:TRANsform:DISTance:UNIT FEET
```

Front Panel Access: **Shift 8** (System), Application Options, Units

Get Distance List

:CALCulate<Tr>:TRANsform:DISTance:DATA?

Description: Produces the distance list in meters for the given trace. `<Tr>` is the trace number in the range 1 to 8 (1 to 4 for Traces TR1 to TR4 and 5 to 8 for Memory M1 to M4). If no trace number is specified, then the `<Tr>` parameter defaults to trace number 1. The response begins with an ASCII header that specifies the number of data bytes. It appears in the format `#AX,` where `A` is the number of digits in `X,` and `X` is the number of bytes that follow the header. Each distance value is returned in scientific notation and separated by a comma delimiter.

Syntax: `:CALCulate<Tr>:TRANsform:DISTance:DATA?`

Cmd Parameter: **NA** (query only)

Query Response: `<char> returns block data (meters)`

Default Unit: Meters

Front Panel Access: **NA**

3-9 :Display Subsystem

This subsystem provides commands that modify the display of data for the user. They do not modify the way in which data are returned to the controller.

Trace Display

:DISPlay[:WINDow]:TRACe TRACe|MEMory|BOTH

Description: Sets the display type for the current active trace. Setting the display type to TRAC will display the trace only. Setting the display type to MEM will display the trace memory only. Setting the display type to BOTH will display both the trace and memory.

Syntax:
```
:DISPlay[:WINDow]:TRACe TRACe|MEMory|BOTH
:DISPlay[:WINDow]:TRACe?
```

Cmd Parameter: `<char> TRACe|MEMory|BOTH`

Query Response: `<char> TRAC|MEM|BOTH`

Default Value: TRAC

Front Panel Access: **Shift 5** (Trace), Display

Trace Format

:DISPlay[:WINDow]:TRACe:FORMat

Description: Defines the display trace format. The query version of this command returns “SING” if the trace format is set to Single, “DUAL” if set to dual, “TRI” if set to Tri, and “QUAD” if set to Quad.

Syntax:
```
:DISPlay[:WINDow]:TRACe:FORMat SINGle|DUAL|TRI|QUAD
:DISPlay[:WINDow]:TRACe:FORMat?
```

Cmd Parameter: `<char> SINGle|DUAL|TRI|QUAD`

Query Response: `<char> SING|DUAL|TRI|QUAD`

Default Value: QUAD

Example: To set the display trace format to Dual:

```
:DISPlay:TRACe:FORMat DUAL
```

Front Panel Access: **Measure**, Trace Format

Group Delay Aperture

:DISPlay:WINDow:TRACe:Y[:SCALe]:GDAPerture

Description: Sets the Group Delay aperture value (which is common to all traces). The query version of this command produces the Group Delay aperture as its output.

Syntax:
```
:DISPlay:WINDow:TRACe:Y[:SCALe]:GDAPerture <integer>
:DISPlay:WINDow:TRACe:Y[:SCALe]:GDAPerture?
```

Cmd Parameter: `<NR1> <integer>`

Query Response: `<NR1> <integer>`

Default Value: 2

Range: 2 to 20

Front Panel Access: **Scale**, Aperture

Note	Graph type must be Group Delay in order to display the Aperture % soft key in the Scale menu.

Scale Resolution Per Division

:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:PDIVision

Description: Sets the scale per division for the y-axis. For Group Delay, sets the scale (time/division) for the y-axis. For Phase, sets the scale (degree/division) for the y-axis. For Log Magnitude and Log Mag/2, sets the scale (dB/division) for the y-axis. For all other measurements, the y-axis is unitless. `<Tr>` is the trace number in the range 1 to 4. If no trace number is specified, then the `<Tr>` parameter defaults to trace number 1.

Syntax:
```
:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:PDIVision <value>
:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:PDIVision?
```

Cmd Parameter: `<NRf> <value> (depends on display type)`

Query Response: `<NR3> <value> (depends on display type)`

Default Value:
Log Magnitude: 10 dB
Phase: 45°
SWR: 1
Group Delay: 1 ns
Real: 0.2
Imag: 0.2
Log Mag/2: 10 dB
Smith Chart: 10
Log Polar: 10 dB
Linear Polar: 0.2
Real Impedance: 10 ohm
Imaginary Impedance: 10 ohm

Default Unit: Current active value unit (For time, the default for setting is seconds, but the query is always returned in nanoseconds (ns).)

Range:
Log Magnitude: 0.05 dB to 40 dB
Phase: 0.1° to 90°
SWR: 0.001 to 10
Group Delay: 1 ps to 260 ns
Real: 0.01 to 260
Imag: 0.01 to 260
Log Mag/2: 0.05 dB to 40 dB
Smith Chart: 1 to 260
Log Polar: 0.05 dB to 40 dB
Linear Polar: 0.001 to 26
Real Impedance: 0.01 ohm to 260 ohm
Imaginary Impedance: 0.01 ohm to 260 ohm

Note Although these values are not used for Smith Chart, when you query or set through SCPI, the instrument always returns a value. For Smith Chart, use `:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:SMCHart 0|10|20|30|-3`

Front Panel Access: **Scale**, Resolution Per Div

Scale Reference Level

:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:RLEVel

Description: Sets the reference level scale value for the y-axis. <Tr> is the trace number in the range 1 to 4. If no trace number is specified, then the <Tr> parameter defaults to trace number 1.

Syntax:
```
:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:RLEVel <value>
:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:RLEVel?
```

Cmd Parameter: <NR3> <value> (depends on display type)

Query Response: <NR3> <value> (depends on display type)

Default Value:
Log Magnitude: 0 dB
SWR: 1
Phase: 0°
Group Delay: 0 ps
Real: 0
Imag: 0
Log Mag/2: 0 dB
Smith Chart: 10
Log Polar: 0 dB
Linear Polar: 1
Real Impedance: 50 ohm
Imaginary Impedance: 0 ohm

Default Unit: Current active value unit (For time, the default for setting is seconds, but the query is always returned in nanoseconds (ns).)

Range:
Log Magnitude: −120 dB to +120 dB
SWR: 1 to 10
Phase: −180° to +180°
Group Delay: 0 ps to 260 ns
Real: −10000 to +10000
Imag: −10000 to +10000
Log Mag/2: −120 dB to +120 dB
Smith Chart: 1 to 260
Log Polar: −120 dB to +120 dB
Linear Polar: 0.005 to 130
Real Impedance: −100000 ohm to +1000000 ohm
Imaginary Impedance: −100000 ohm to +1000000 ohm

Note	Although these values are not used for Smith Chart, when you query or set through SCPI, the instrument always returns a value. For Smith Chart, use `:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:SMCHart 0|10|20|30|-3`

Related Command: `:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:RPOSition`

Front Panel Access: **Scale**, Reference Value

Scale Reference Line

:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:RPOSition

Description: Sets the reference line scale value for the y-axis.

Syntax:
```
:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:RPOSition <integer>
:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:RPOSition?
```

Cmd Parameter: `<NR1> <integer>`

Query Response: `<NR1> <integer>`

Default Value:
Log Magnitude: 9
SWR: 1
Phase: 5
Group Delay: 5
Real: 5
Imag: 5
Log Mag/2: 9
Smith Chart: 10
Real Impedance: 5 ohm
Imaginary Impedance: 5 ohm

Range:
Log Magnitude: 0 to 10
SWR: 0 to 10
Phase: 0 to 8
Group Delay: 0 to 10
Real: 0 to 10
Imag: 0 to 10
Log Mag/2: 0 to 10
Smith Chart: 0 to 10
Real Impedance: 0 ohm to 10 ohm
Imaginary Impedance: 0 ohm to 10 ohm

Note	Although these values are not used for Smith Chart, when you query or set through SCPI, the instrument always returns a value. For Smith Chart, use `:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:SMCHart 0|10|20|30|-3`

Related Command: `:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:RLEVel`

Front Panel Access: **Scale**, Reference Line

Smith Chart Scalable Type

:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:SMCHart

Description: Sets the Smith chart display scale type of the given trace number specified by <Tr>. <Tr> is the trace number in the range 1 to 4. If no trace number is specified, then the <Tr> parameter defaults to trace number 1. Setting the value to 0 is equivalent of setting the Smith Chart scale to "Normal" on the front panel. Setting the value to 10 is equivalent of setting the Smith Chart scale to "Expand 10 dB" on the front panel. Setting the value to 20 is equivalent of setting the Smith Chart scale to "Expand 20 dB" on the front panel. Setting the value to 30 is equivalent of setting the Smith Chart scale to "Expand 30 dB" on the front panel. Setting the value to –3 is equivalent of setting the Smith Chart scale to "Compress 3 dB" on the front panel.

Syntax:
```
:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:SMCHart
0|10|20|30|-3
:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:SMCHart?
```

Cmd Parameter: `<char> 0|10|20|30|-3`

Query Response: `<char> 0|10|20|30|-3`

Default Value: 0 (Normal)

Front Panel Access: **Scale**

3-10 :Format Subsystem

This subsystem contains commands that determine the formatting of numeric data when it is transferred.

The format setting affects data in specific commands only. If a command is affected, it will be noted in the command description.

Numeric Data Format

:FORMat[:READings][:DATA]

Description: This command specifies the format in which data is returned in certain commands. `ASCii` format returns the data in comma-separated ASCII format. The units are the current instrument units. This format requires many more bytes, so it will be the slowest format.

`INTeger,32` values are signed 32-bit integers in little-endian byte order. This format returns the data in 4-byte blocks.

`REAL,32` values are 32-bit floating point numbers conforming to the IEEE 754 standard in little-endian byte order. This format returns the data in 4-byte binary format. The units are the current instrument units.

Both `INTeger` and `REAL` formats return a definite block length. Each transfer begins with an ASCII header, such as #42204 for INTeger,32 and REAL,32. The first digit represents the number of following digits in the header (in this example, 4).

The remainder of the header indicates the number of bytes that follow the header (in this example, 2204 for INT,32 and REAL,32). You then divide the number of following bytes by the number of bytes in the data format that you have chosen (4 for both INTeger,32 and REAL,32…so 2204/4) to get the number of data points (in this example, 551).

Syntax:
```
:FORMat[:READings][:DATA] ASCii|INTeger,32|REAL,32
:FORMat[:READings][:DATA]?
```

Cmd Parameter: `<char> ASCii|INTeger,32|REAL,32`

Query Response: `<char> ASC|INT,32|REAL,32`

Default Value: ASC

Related Command: `:TRACe[:DATA]`

Front Panel Access: **NA**

3-11 :INITiate Subsystem

This subsystem controls the triggering of measurements.

Continuous/Single Sweep

`:INITiate:CONTinuous`

Description: Sets the sweep to continuous. If the instrument is currently on hold, and if sweep type is set to continuous, then setting to ON restarts the sweep. If the instrument is currently on hold, and if sweep type is set to single, then setting to ON will set the Sweep Type to Continuous and restart the sweep. If the instrument is currently sweeping, then setting a value of OFF or 0 sets the Sweep Type to Single and holds the sweep. The default value is ON. That is, sending `:INIT:CONT` is equivalent to sending `:INIT:CONT ON.` The query version of this command returns a `1` if the instrument is set to Continuous and Run, or it returns a `0` if set to Hold.

Syntax:
```
:INITiate:CONTinuous OFF|ON|0|1
:INITiate:CONTinuous?
```

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: ON or 1 (query returns 1 for ON)

Related Command: `:INITiate:HOLD`

Front Panel Access: **Shift 3** (Sweep), Run/Hold

Hold Sweep

:INITiate:HOLD

Description: Sets the sweep to hold. If the instrument is currently sweeping, then setting a value of ON or 1 pauses the sweep. If the instrument is currently not sweeping, and if sweep type is set to continuous, then setting a value of OFF or 0, restarts the sweep. If the instrument is currently not sweeping, and if sweep type is set to single, then setting a value of OFF or 0, triggers a sweep. The query version of the command returns a `1` if the hold command is set, and it returns a `0` if a Run is set.

Syntax:
```
:INITiate:HOLD OFF|ON|0|1
:INITiate:HOLD?
```

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: OFF or 0 (query returns 0 for OFF)

Related Command: `:INITiate:CONTinuous`

Front Panel Access: **Shift-3** (Sweep), Run/Hold

Trigger Sweep/Measurement

:INITiate[:IMMediate]

Description: Initiates a sweep/measurement. Use this command in combination with `:STATus:OPERation?` to synchronize the capture of one complete set of data. When this command is sent, the "sweep complete" bit of `:STATus:OPERation?` is set to `0`, indicating that the measurement is not completed. The data collection is then triggered. The controlling program can poll `:STATus:OPERation?` to determine the status. When the "sweep complete" bit is set to `1`, data is ready to be retrieved.

If sweep is set to Run, and sweep type is set to Continuous, then sending the `:INIT:IMM` command restarts the sweep.

If sweep is set to Hold, and sweep type is set to Single, then sending the `:INIT:IMM` command starts a sweep (instrument is temporarily in Run). After a single sweep is completed, the instrument returns to Hold.

Syntax: `:INITiate[:IMMediate]`

Cmd Parameter: **NA**

Query Response: **NA** (no query)

Related Command: `:STATus:OPERation?`

Front Panel Access: **NA**

3-12 :INPut Subsystem

This subsystem controls characteristics of the input port.

Internal Bias Tee Current

:INPut<port_no>:BIAS:CURRent

Description: Sets the internal bias tee current limit for the specified port. When this limit is exceeded, the Bias Tee trips (turns OFF). `<port_no>` is the specified internal bias tee port number 2. The query version of this command returns either the measured internal current or the set internal current limit (both are returned in milliampere units). To return the measured internal current for the specified port number, send the query command either with no value specified after the "?" (default condition) or with a value of `0` specified after the "?". Note that the query result for the measured internal current is valid only if the Bias Tee state is set to internal. If a value of `1` is specified after the "?", then the query version of this command returns the internal current limit that is set for the given port number.

Syntax:
```
:INPut<port_no>:BIAS:CURRent <current>
:INPut<port_no>:BIAS:CURRent? [0|1]
```

Cmd Parameter: `<NRf> <current> (milliampere)`

Query Response : `<char> [0|1] (returns value in milliampere)`

Default Value: 450 mA when querying the internal current limit (`:INPut<port_no>:BIAS:CURRent? 1`). The default value for querying the measured current depends upon what is connected to the port.

Default Unit: milliampere (mA)

Range: 0 mA to 450 mA

Front Panel Access: **Shift-3** (Sweep), Configure Ports, Bias Tee Setup, Int Current Limit P2

Internal Bias Tee Tripped State

:INPut:BIAS:INTernal:TRIPped[:STATe]?

Description: Returns whether the internal bias tee is tripped. Returns `1` for tripped, otherwise returns `0`.

Syntax: `:INPut:BIAS:INTernal:TRIPped[:STATe]?`

Cmd Parameter: **NA** (query only)

Query Response : `<boolean> [0|1]`

Front Panel Access: **NA**

Bias Tee State

:INPut:BIAS:STATe

Description: Enables or disables the bias tee. Query returns OFF|INT. For OFF state, query returns OFF (not 0).

Syntax:
```
:INPut:BIAS:STATe OFF|INTernal
:INPut:BIAS:STATe?
```

Parameter: `OFF|INTernal`

Cmd Parameter: `<char> OFF|INTernal`

Query Response: `<char> OFF|INT`

Default Value: OFF

Front Panel Access: **Shift-3** (Sweep), Configure Ports, Bias Tee Setup, Bias Tee

Internal Bias Tee Voltage

:INPut<port_no>:BIAS:VOLTage

Description: Sets the voltage of the internal bias tee for the specified port number. `<port_no>` is the specified internal bias tee port number (currently, only port 2 is supported). The query version of this command returns either the measured internal bias tee voltage or the set internal bias tee voltage (both are returned in Volts). Note that the measured voltage can be slightly different than the set voltage depending on the load conditions. To return the last measured internal bias tee voltage for the specified port number, send the query command either with no value specified after the "?" (default condition) or with a value of 0 specified after the "?". Note that the query result for the measured internal bias tee voltage is valid only if the Bias Tee state is set to internal. If a value of 1 is specified after the "?", then the query version of this command returns the internal bias tee voltage that was set for the given port number.

Syntax:
```
:INPut<port_no>:BIAS:VOLTage <voltage>
:INPut<port_no>:BIAS:VOLTage? [0|1]
```

Cmd Parameter: `<NRf> <12 to 32 Volts>`

Query Response: `<NR3> <Volts>`

Default Value: +12 V when querying the internal bias tee voltage that was set (`:INPut<port_no>:BIAS:VOLTage? 1`). The default value for querying the last measured bias tee voltage may be slightly different depending upon what is connected to the port.

Default Units: Volts

Range: 12 V to 32 V

Front Panel Access: **Shift-3** (Sweep), Configure Ports, Bias Tee Setup, Int Voltage P2

3-13 :MMEMory Subsystem

The Mass :MEMory subsystem contains functions that provide access to the instrument setup and data storage.

Table 3-4. :MMEMory Subsystem

Keyword	Parameter Form	Parameter Data or Units	Notes
`:MMEMory`			
`:LOAD`		Refer to ":MMEMory:LOAD Subsystem" on page 3-59	
`:STORe`		Refer to ":MMEMory:STORe Subsystem" on page 3-61	

3-14 :MMEMory:LOAD Subsystem

The Mass MEMory LOAD subsystem contains commands to transfer from the mass memory device to the internal memory.

Recall Setup

:MMEMory:LOAD:STATe

Description: Recalls a previously stored setup from the current storage location. The saved setup that is to be loaded is specified by `<filename>`. `<filename>` must be enclosed in either single quotes (' ') or double quotes (" ") and must include the extension `".stp"`. The `<integer>` parameter is not currently used, but it must be sent. Send a value of `1`.

Note	When recalling setups for modes other than the current mode, Anritsu recommends that you first change modes, then recall the setup. Use ":INSTrument:NSELect" on page 7-3 or ":INSTrument[:SELect]" on page 7-4 to change the instrument state, then use `:MMEMory:LOAD:STATe` to recall the setup.

Syntax: `:MMEMory:LOAD:STATe <integer>, <filename>`

Cmd Parameter: `<integer>, <string> (1, file name)`

Query Response: **NA** (no query)

Related Command: `:MMEMory:STORe:STATe`
`:MMEMory:MSIS INTernal|USB`

Front Panel Access: **Shift-7** (File), Recall

Recall Measurement

:MMEMory:LOAD:TRACe

Description: Recalls a previously stored measurement trace from the current storage location. The saved measurement trace that is to be loaded is specified by `<filename>`. `<filename>` must be enclosed in either single quotes (‘ ’) or double quotes (“ ”) and must contain a file extension of `".mna"`. Note that the trace that is specified by `<filename>` must be available at the current mass storage location. The `<integer>` parameter is not currently in use, but it must be sent. Send a `1`. File Extensions: `".mna"`.

Syntax: `:MMEMory:LOAD:TRACe <integer>, <filename>`

Cmd Parameter: `<integer>, <string> (1, file name)`

Query Response: **NA** (no query)

Example: To recall trace with file name “trace”:

```
:MMEMory:LOAD:TRACe 1,"trace.mna"
```

Related Command: `:MMEMory:STORe:TRACe`
`:MMEMory:MSIS INTernal|USB`

Front Panel Access: **Shift-7** (File), Recall

3-15 :MMEMory:STORe Subsystem

The Mass MEMory STORe subsystem contains commands to transfer from the internal memory to the mass memory device.

Save Setup

:MMEMory:STORe:STATe

Description: Stores the current setup into the file that is specified by `<filename>`. `<filename>` must be enclosed in either single quotes (‘ ’) or double quotes (“ ”) and must not contain a file extension. The `<integer>` is used to distinguish whether the calibration should be saving with the setup. Send a 1 to save setup without a calibration. Send a 2 to save setup with calibration.

Syntax: `:MMEMory:STORe:STATe <integer>, <filename>`

Cmd Parameter: `<integer>, <string> (1|2, filename)`

Query Response: **NA** (no query)

Front Panel Access: **NA**

Save Measurement

:MMEMory:STORe:TRACe

Description: Stores the trace into the file that is specified by `<filename>`. `<filename>` must be enclosed in either single quotes (‘ ’) or double quotes (“ ”) and must not contain a file extension. Note that existing files of the same name will not be overwritten. The `<integer>` parameter is used to distinguish which type of files to save. The following types are available:

<Integer>	:	File type
1	:	Measurement file (default, if number is not 1 to 6)
2	:	S2P Real/Imag
3	:	S2P Lin Mag/Phase
4	:	S2P Log Mag/Phase
5	:	Text
6	:	CSV

Syntax: `:MMEMory:STORe:TRACe <integer>, <filename>`

Cmd Parameter: `<integer>, <string> (1|2, filename)`

Query Response: **NA** (no query)

Example: To save the trace into the file named “trace”.

```
:MMEMory:STORe:TRACe 1,”trace”
```

Related Command: `:MMEMory:LOAD:TRACe`

Front Panel Access: **Shift-7** (File), Save
Shift-7 (File), Save Measurement

3-16 [:SENSe] Subsystem

The commands in this subsystem relate to device-specific parameters, not to signal-oriented parameters.

Table 3-5. [:SENSe] Subsystem

Keyword	Parameter Data or Units
`[:SENSe]`	
`:APPLication`	Refer to "[:SENSe]:APPLication Subsystem" on page 3-64
`:AVERage`	Refer to "[:SENSe]:AVERage Subsystem" on page 3-65
`:CALibration`	Refer to "[:SENSe]:CALibration Subsystem" on page 3-66
`:CORRection`	Refer to "[:SENSe]:CORRection Subsystem" on page 3-67
`:FREQuency`	Refer to "[:SENSe]:FREQuency Subsystem" on page 3-90
`:SWEep`	Refer to "[:SENSe]:SWEep Subsystem" on page 3-95
`:TRACe`	Refer to "[:SENSe]:TRACe Subsystem" on page 3-97

RF Power In Hold State

`[:SENSe]:RFON[:STATe]`

Description: Sets the state of the RF output signal at the VNA ports to be either ON or OFF when the sweep is set to Hold. When set to ON, the RF signal continues to be energized when the sweep is in hold. When set to OFF, the RF signal is turned off during the hold condition. Note that the sweep may require more time to stabilize when it is set to run, if the RF has been turned OFF during hold.

Syntax: `[:SENSe]:RFON[:STATe] ON|OFF|1|0`

Cmd Parameter: `<boolean> ON|OFF|1|0`

Query Response: `<bNR1> 1|0`

Default Value: ON

Front Panel Access: **Shift-3** (Sweep), RF Pwr in Hold

3-17 [:SENSe]:APPLication Subsystem

This subsystem contains application specific commands.

Application Self Test

[:SENSe]:APPLication:TST?

Description: Executes an application self test and reports whether any errors were detected. A return value of "0" indicates that the test was completed without detecting any error.

Two self test types can be specified. If no test type is specified, then the test defaults to NORMal. The PWRon self test is a scaled-down version of the normal self test that runs during the instrument power-on cycle.

Syntax: `[:SENSe]:APPLication:TST? NORMal|PWRon`

Cmd Parameter: **NA** (query only)

Query Parameter: `<char> NORMal|PWRon`

Query Response: `<NR1> <integer>`

Front Panel Access: **NA**

Application Self Test Result

[:SENSe]:APPLication:TST:RESult?

Title: Application Self Test Result

Description: Returns the application self test result of the previous call to the application self test. The response begins with an ASCII header. The header specifies the number of following bytes. It appears in the format `#AX<block data>`, where `A` is the number of digits in `X`, and `X` is the number of bytes that follow the header. The first information of the <block data> contains the overall self test string (`"PASSED"` or `"FAILED"`) followed by a comma, and each self test result separated by a comma. Each subset of the result is included in angle brackets, `< >`.

Note that an application self test command must be called prior to calling this command in order for the result to be valid.

Syntax: `[:SENSe]:APPLication:TST:RESult?`

Cmd Parameter: **NA** (query only)

Query Response: `<block> (No units, NA)`

Front Panel Access: **NA**

3-18 [:SENSe]:AVERage Subsystem

This subsystem contains commands that are related to the combination of the data from consecutive sweeps. Use commands in this subsystem to control sweep-to-sweep averaging and max hold functionality.

Restart Averaging

[:SENSe]:AVERage:CLEar

Description: No query. Clears and restarts averaging of the measurement data. Note that sweep averaging count must be set to greater than 1 for averaging to restart.

Syntax: `[:SENSe]:AVERage:CLEar`

Cmd Parameter: `<char>`

Query Response: **NA** (no query)

Front Panel Access: **NA**

Number of Traces to Average

[:SENSe]:AVERage:COUNt

Description: Sets the number of traces to be averaged. Note that when averaging count is set to be greater than 1, sweep averaging is turned on. To stop Syntax

```
[:SENSe]:AVERage:COUNt <integer>
[:SENSe]:AVERage:COUNt?
```

Cmd Parameter: `<NR1> <integer>`

Query Response: `<NR1> <integer>`

Default Value: 1

Range: 1 to 65535

Front Panel Access: **Shift-3** (Sweep), Sweep Averaging

3-19 [:SENSe]:CALibration Subsystem

This subsystem controls the system calibration.

Calibration State

[:SENSe]:CALibration:STATe?

Description: Reports the calibrated state. This command returns a 0 if there is no valid calibration, otherwise it returns the bit of the S parameters that has a valid calibration. The bits are as follows:

S11 bit	0x01
S21 bit	0x04

For example, if a value of 5 is returned, then both of the S parameter bits are valid (since decimal 5 is equivalent to binary 0101).

Syntax: `[:SENSe]:CALibration:STATe?`

Cmd Parameter: **NA** (query only)

Query Response: `<NR1> <integer> (0 to 5)`

Front Panel Access: **NA**

3-20 [:SENSe]:CORRection Subsystem

This subsystem provides commands for losses or gains external to the instrument.

Table 3-6. [:SENSe]:CORRection Subsystem

Keyword	Parameter Data or Units
`[:SENSe]`	
`:CORRection`	
`:CKIT`	Refer to "[:SENSe]:CORRection:CKIT Subsystem" on page 3-70
`:COLLect`	Refer to "[:SENSe]:CORRection:COLLect Subsystem" on page 3-78

Error Correction Data

[:SENSe]:CORRection:DATA?

Title: Error Correction Data

Description: Transfers the system error correction data from the instrument to the controller. `<error term parameter>` are string parameters that describe the different error terms.

<error term parameter>	Descriptions
ERF	(Forward) Reflection tracking
EDF	(Forward) Directivity
ESF	(Forward) Source match
ETF	(Forward) Transmission tracking
ELF	(Forward) Load match
EXF	(Forward) Isolation
ETFS	(Forward Sensitivity) Transmission tracking
ELFS	(Forward Sensitivity) Load match
EXFS	(Forward Sensitivity) Isolation

The format of the block data that is returned can be specified by the command `:FORMat:DATA`. The response begins with an ASCII header that specifies the number of data bytes. It appears in the format `#AX`, where `A` is the number of digits in `X`, and `X` is the number of bytes that follow the header. Each data point is separated by a comma delimiter. Each term contains one complex value (real and imaginary) for each sweep point.

Syntax: `[:SENSe]:CORRection:DATA? <error terms parameters>`

Cmd Parameter: **NA** (query only)

Query Parameter: `<char> <error terms parameters>`

Query Response: `<char> (returns block data)`

Related Command: `:FORMat:DATA`

Front Panel Access: **NA**

Smith Chart Reference Impedance

[:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude]:SMCHart

Description: Sets the Smith Chart reference impedance. Sets `50` for 50 ohm. Sets `75` for 75 ohm. The query form of the command returns the current Smith Chart reference impedance in ohms.

Syntax:
```
[:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude]
:SMCHart 50|75
[:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude]
:SMCHart?
```

Cmd Parameter: `<char> 50|75`

Query Response: `<char> 50|75`

Default Value: 50

Range: 50, 75

Front Panel Access: **Scale**, Reference Impedance

Calibration Correction State

[:SENSe]:CORRection[:STATe]

Description: Turns the calibration error correction ON or OFF. Note that error correction can be turned ON only if valid calibration is available.

Syntax:
```
[:SENSe]:CORRection[:STATe] OFF|ON
[:SENSe]:CORRection[:STATe]?
```

Parameter: `OFF|ON`

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: 0

Front Panel Access: **Shift-2** (Calibrate), Cal Correction

3-21 [:SENSe]:CORRection:CKIT Subsystem

This subsystem provides commands that modify and configure the device under test (DUT).

Table 3-7. [:SENSe]:CORRection:CKIT Subsystem

Keyword	Parameter Data or Units
[:SENSe]	
:CORRection	
:CKIT	
:USER{1-4}	Refer to "[:SENSe]:CORRection:CKIT:USER Subsystem" on page 3-75

Calibration Connector Information

[:SENSe]:CORRection:CKIT:INFormation?

Description: Returns a string of information of the given calibration connector. <connector> defines the connector family and can be given in either long or short form. [connector-name] is a string that defines the name that is associated with the given <connector> and is optional. [connector-name] must be enclosed by parentheses. Note that the connector must be valid for the current calibration line type.

Note that user 1, user 2, user 3, or user 4 is based on the current calibration method.

The query response begins with an ASCII header. The header specifies the number of following bytes. It appears in the format #AX, where A is the number of digits in X, and X is the number of bytes that follow the header. Parameters are returned in comma-delimited ASCII format. Each parameter is returned as "NAME=VALUE[UNITS]".

The tables below list the available connectors and connector names that are associated with the calibration line types. Note that Coax User cal kits do not have a calibration name associated with them.

COAX <connector>	[connector-name]	Description
NMALe	OSLN50	N-Conn(M) Cal Kit: OSLN50. If no connector-name is given, information will default to this. Query Response: <block> (returns comma-delimted ASCII format)
NMALe	SLN50A or OSLN50A-8 or OSLN50A-18	N-Conn(M) Cal Kit: OSLN50A-8 or OSLN50A-18 Query Response: <block> (returns comma-delimted ASCII format)
NMALe	TOSLN50A or TOSLN50A-8 or TOSLN50A-18	N-Conn(M) Cal Kit: TOSLN50A-8 or TOSLN50A-18 Query Response: <block> (returns comma-delimted ASCII format)
NFEMale	OSLNF50	N-Conn(F) Cal Kit: OSLNF50. If no connector name is given, then information will default to this. Query Response: <block> (returns comma-delimted ASCII format)
NFEMale	OSLNF50A or OSLNF50A-8 or OSLNF50A-18	N-Conn(F) Cal Kit: OSLNF50A-8 or OSLNF50A-18 Query Response: <block> (returns comma-delimted ASCII format)
NFEMale	TOSLNF50A or TOSLNF50A-8 or TOSLNF50A-18	N-Conn(F) Cal Kit: TOSLNF50A-8 or TOSLNF50A-18 Query Response: <block> (returns comma-delimted ASCII format)

COAX <connector>	[connector-name]	Description
KMALe	OSLK50	K-Conn(M) Cal Kit: OSLK50. If no connector name is given, then information will default to this. Query Response: <block> (returns comma-delimted ASCII format)
KMALe	TOSLK50A or TOSLK50A-20	K-Conn(M) Cal Kit: TOSLK50A-20 Query Response: <block> (returns comma-delimted ASCII format)
KFEMale	OSLKF50	K-Conn(F) Cal Kit: OSLKF50 If no connector-name is given, then information will default to this. Query Response: <block> (returns comma-delimted ASCII format)
KFEMale	TOSLKF50A or TOSLKF50A-20	K-Conn(F) Cal Kit: TOSLKF50A-20 Query Response: <block> (returns comma-delimted ASCII format)
716Male	2000-767	7/16(M) Cal Kit: 2000-767. If no connector-name is given, then information will default to this. Query Response: <block> (returns comma-delimted ASCII format)
716Male	2000-1618 or 2000-1618-R	7/16(M) Cal Kit: 2000-1618-R Query Response: <block> (returns comma-delimted ASCII format)
716Female	2000-768	7/16(F) Cal Kit: 2000-768 If no connector name is given, then information will default to this. Query Response: <block> (returns comma-delimted ASCII format)

COAX <connector>	[connector-name]	Description
716Female	2000-1619 or 2000-1619-R	7/16(F) Cal Kit: 2000-1619-R Query Response: <block> (returns comma-delimted ASCII format)
TNCMale		TNC(M) Cal Kit: 1091-5x & 1015-55. If no connector-name is given, then information will default to this. Query Response: <block> (returns comma-delimted ASCII format)
TNCFemale		TNC(F) Cal Kit: 1091-5x & 1015-54 If no connector-name is given, then information will default to this. Query Response: <block> (returns comma-delimted ASCII format)
SMAMale	3650	SMA(M) Cal Kit: 3650
SMAFemale	3650	SMA(F) Cal Kit: 3650
431Female	2000-1914-R	COAX DUT 4.3-10 Female
431Male	2000-1915-R	COAX DUT 4.3-10 Male
USR1		User 1 cal Kit information for the current calibration method. Query Response: <block> (returns comma-delimted ASCII format)
USR2		User 1 cal Kit information for the current calibration method. Query Response: <block> (returns comma-delimted ASCII format)
USR3		User 1 cal Kit information for the current calibration method. Query Response: <block> (returns comma-delimted ASCII format)

COAX <connector>	[connector-name]	Description
USR4		User 1 cal Kit information for the current calibration method. Query Response: <block> (returns comma-delimted ASCII format)

Syntax: `[:SENSe]:CORRection:CKIT:INFormation? <connector>, [connector-name string]`

Cmd Parameter: **NA** (query only)

Query Parameter: `<char> <connector>, [connector-name string]`

Query Response: `<block> (returns comma-delimted ASCII format)`

Example: To get information for K-Conn(M) with cal Kit TOSLK50A-20:

```
:SENS:CORR:CKIT:INF? KMAL, "TOSLK50A”
```

or

```
:SENS:CORR:CKIT:INF? KMAL, "TOSLK50A-20"
```

The response for this query when using SOLT COAX:

```
#6000164OPEN=5.010 mm,SHORT=5.010 mm,
THRU=16.070 mm,C0=4.500 e-15,C1=395.000 e-27,
C2=-20.000 e-36,C3=0.400 e-45,L0=4.000 e-12,
L1=-650.000 e-24,L3=39.000 e-33,L3=-0.640 e-42
```

Front Panel Access: **Shift-2** (Calibrate), DUT Port Setup, DUT Port 1

3-22 [:SENSe]:CORRection:CKIT:USER Subsystem

This subsystem contains commands to configure the user device under test (DUT).

DUT User Inductance Coefficient value

[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX:SOLT:L0|1|2|3

Description: Sets the DUT inductance value for the specified Short component of the user-defined SOLT Cal Kit.

Syntax: `[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX:SOLT:L0|1|2|3 <inductance>`

`[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX:SOLT:L0|1|2|3?`

Cmd Parameter: `<NRf> <inductance> (e-12, e-24, e-34, e-42)`

Query Response: `<NR3> <inductance> (e-12, e-24, e-34, e-42)`

Example: To set the DUT inductance L1 for User 1 with line type coax to 5:

`:SENS:CORR:CKIT:USER:COAX:SOLT:L1 5`

To query the DUT inductance L1 for User 1:

`:SENS:CORR:CKIT:USER:COAX:SOLT:L1?`

The query response would be: `5.000`

Front Panel Access: **Shift-2** (Calibrate), DUT Port Setup, Setup User-Defined, Short

DUT User Capacitance Coefficient value

[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX:SOLT:C0|1|2|3

Description: Sets the DUT capacitance value for the specified user.

Syntax: `[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX:SOLT:C0|1|2|3 <capacitance>`
`[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX:SOLT:C0|1|2|3?`

Cmd Parameter: `<NRf> <capacitance> (e-15, e-27, e-36, e-45)`

Query Response: `<NR3> <capacitance> (e-15, e-27, e-36, e-45)`

Example: To set the DUT capacitance #1 for User 1 with line type coax to 5:

`:SENS:CORR:CKIT:USER:COAX:SOLT:C1 5`

Front Panel Access: **NA**

DUT User Name

[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX:SOLT:NAME

Description: Sets the DUT name for the specified user.

Syntax:
```
[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX
:SOLT:NAME <string>
[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX
:SOLT:NAME?
```

Cmd Parameter: `<string> (no parameter data or units)`

Query Response: `<string> (no parameter data or units)`

Example: To set the DUT name for User 1 with line type coax and calibration method SOLT:

```
:SENS:CORR:CKIT:USER:COAX:SOLT:NAME "SOLT1"
```

Front Panel Access: **NA**

DUT User Open Offset

[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX:SOLT:OPEN

Description: Sets the DUT capacitance value for the specified user.

Syntax:
```
[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX:SOLT:OPEN
<length>
[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX:SOLT:OPEN?
```

Cmd Parameter: `<NRf> <length> (millimeters)`

Query Response: `<NR3> <length> (millimeters)`

Example: To set the DUT open offset for User 1 with line type coax to 3 mm:

```
:SENS:CORR:CKIT:USER:COAX:SOLT:OPEN 3
```

Front Panel Access: **NA**

DUT User Short Offset for calibration method SOLT

[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX:SOLT:SHORt

Description: Sets the DUT capacitance value for the specified user.

Syntax:
```
[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX:SOLT:SHORt
<length>
[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX:SOLT:SHORt?
```

Cmd Parameter: `<NRf> <length> (millimeters)`

Query Response: `<NR3> <length> (millimeters)`

Example: To set the DUT short offset for User 1 with line type coax to 3 mm:

```
:SENSe:CORRection:CKIT:USER:COAX:SOLT:SHORt 3
```

Front Panel Access: **NA**

3-23 [:SENSe]:CORRection:COLLect Subsystem

This subsystem controls the system calibration. To properly perform a calibration, several parameters must be set. The table below lists all of the required commands. First, use the `:MEDium` and `:CONNector` subcommands to specify the calibration line type and the DUT port setup. Then use the `:METHod` and `:TYPE` subcommands to specify the calibration method and the calibration type. Then use the `:ACQUire` subcommand to specify the calibration components to be measured. Finally, use the `:SAVe` subcommand to calculate, save, and finish the calibration. Note that the calibration components do not need to be measured in any specific order.

Calibration Abort

`[:SENSe]:CORRection:COLLect:ABORt:ALL`

Description: Aborts the calibration measurement and restarts the current sweep or measurement, or both.

Syntax: `[:SENSe]:CORRection:COLLect:ABORt:ALL`

Cmd Parameter: **NA**

Query Response: **NA** (no query)

Front Panel Access: **NA**

Calibration Steps

[:SENSe]:CORRection:COLLect[:ACQUire]

Description: Performs a measurement of the given steps. `<cal steps>` is the calibration step to be performed and must be one of the following values:

`OPEN|SHORT|LOAD|`
`THRU|ISOLation`

`<port_no>` is the port number, 1. For calibration step `OPEN, SHORT, and LOAD,` valid port number is 1. For calibration step `THRU` and `ISOLation,` valid port number is 1 for Fwd.

Note that the calibration step must be valid for the given calibration type and calibration method. Refer to Table 3-8 for a list of valid calibration steps for each type and method.

The query version of this command returns a string that consists of the last calibration step measurement that was performed followed by the port number. The calibration step and port number are delimited by a comma. Note that if no calibration step has been processed, then this command returns the string `"NONE, 0"`.

Syntax: `[:SENSe]:CORRection:COLLect[:ACQUire] <cal steps>, <port_no>`
`[:SENSe]:CORRection:COLLect[:ACQUire]?`

Cmd Parameter: `<char> <cal steps>, <port_no>`

Query Parameter: `<char> <cal steps>, <port_no>`

Query Response: `<string>`

Front Panel Access: **Shift-2** (Calibrate), Start Cal

Calibration Steps and Calibration Types:

For each calibration Type, Table 3-8 lists the allowable calibration steps and port_no to be used in command `[:SENSe]:CORRection:COLLect[:ACQUire] <cal steps>, <port_no>`. The calibration steps are different for each calibration Method, and the port_no is different for each calibration Type. For example, for calibration Type RFP1 and calibration Method SOLT, the allowable `<cal steps>, <port_no>` settings are **"OPEN,1"**, "**SHORT,1**", and "**LOAD,1**". The steps that are not allowed are indicated by "—".

Table 3-8. SOLT Calibration Method

Calibration Type	OPEN	SHORT	LOAD	THRU	ISOL
RFP1 (Full S_{11} - Port 1)	1	1	1	—	—
TRFP (Response S_{21} - Trans Response Fwd Path)	—	—	—	1	1
RRP1 (Response S_{11} - Reflection Response Port 1)	1	1	1	—	—
2PFP (1P2P S_{11}, S_{21} - 1 Path 2 Port Fwd Path)	1	1	1	1	1

Calibration Step Status

[:SENSe]:CORRection:COLLect:ACQUire:STATus?

Description: This command requests information about the current calibration step or the specified calibration step. If no calibration step is specified, then it returns a 1 if the current calibration step has completed, otherwise it returns a 0. `<cal steps>` is the calibration step to be performed and must be one of the following values:

`OPEN|SHORT|LOAD|THRU|ISOLation`

`<port_no>` is port number 1. For step `OPEN, SHORT,` valid port number is 1. For calibration step `THRU` and `ISOLation,` valid port number is 1 for Fwd.

Syntax: `[:SENSe]:CORRection:COLLect:ACQUire:STATus? [<cal steps>, <port_no>]`

Cmd Parameter: **NA** (query only)

Query Parameter: `<char> [<cal steps>, <port_no>]`

Query Response: `<NR1> <integer>`

Front Panel Access: **NA**

DUT Port Setup

[:SENSe]:CORRection:COLLect:CONNector<port_no>

Description: Sets the connector family for the given port number. `<port_no>` is the port number, 1. `<connector>` defines the connector family and can be given in either long or short form. `[connector-name]` is a string that defines the name that is associated with the given `<connector>` and is optional. `[connector-name]` must be enclosed by parentheses.

Note that the connector must be valid for the current calibration line type.

(continued)

The tables below list the connector and valid connector name that are associated with the calibration line type COAX. Note that User cal kits do not have a calibration name associated with them.

COAX <connector>	[connector-name]	Description
NMALe	OSLN50	N-Conn(M) Cal Kit: OSLN50. If no connector-name is given, then connector will be set to this. Query Response string: “NMAL”
NMALe	SLN50A or OSLN50A-8 or OSLN50A-18	N-Conn(M) Cal Kit: OSLN50A-8 or OSLN50A-18 Query Response string: “NMAL(OSLN50A-8 or OSLN50A-18)”
NMALe	TOSLN50A or TOSLN50A-8 or TOSLN50A-18	N-Conn(M) Cal Kit: TOSLN50A-8 or TOSLN50A-18 Query Response string: “NMAL(TOSLN50A-8 or TOSLN50A-18)
NFEMale	OSLNF50	N-Conn(F) Cal Kit: OSLNF50. If no connector name is given, then connector will be set to this. Query Response string: “NFEM”
NFEMale	OSLNF50A or OSLNF50A-8 or OSLNF50A-18	N-Conn(F) Cal Kit: OSLNF50A-8 or OSLNF50A-18 Query Response string: “NFEM(OSLNF50A-8 or OSLNF50A-18)”
NFEMale	TOSLNF50A or TOSLNF50A-8 or TOSLNF50A-18	N-Conn(F) Cal Kit: TOSLNF50A-8 or TOSLNF50A-18 Query Response string: “NFEM(TOSLNF50A-8 or TOSLNF50A-18)”
KMALe	OSLK50	K-Conn(M) Cal Kit: OSLK50. If no connector name is given, then connector will be set to this. Query response string: “KMAL”

COAX <connector>	[connector-name]	Description
KMALe	TOSLK50A or TOSLK50A-20	K-Conn(M) Cal Kit: TOSLK50A-20 Query Response string: “KMAL(TOSLK50A-20)”
KFEMale	OSLKF50	K-Conn(F) Cal Kit: OSLKF50 If no connector-name is given, then connector will be set to this. Query response string: “KFEM”
KFEMale	TOSLKF50A or TOSLKF50A-20	K-Conn(F) Cal Kit: TOSLKF50A-20 Query Response string: “KFEM(TOSLKF50A-20)”
716Male	2000-767	7/16(M) Cal Kit: 2000-767. If no connector-name is given, then connector will be set to this. Query response string: “716M”
716Male	2000-1618 or 2000-1618-R	7/16(M) Cal Kit: 2000-1618-R Query Response string: “716M(2000-1618-R)”
716Female	2000-768	7/16(F) Cal Kit: 2000-768 If no connector name is given, then connector will be set to this. Query response string: “716F”
716Female	2000-1619 or 2000-1619-R	7/16(F) Cal Kit: 2000-1619-R Query Response string: “716F(2000-1619-R)”
TNCMale		TNC(M) Cal Kit: 1091-5x & 1015-55. If no connector-name is given, then connector will be set to this. Query response string: “TNCM”.
TNCFemale		TNC(F) Cal Kit: 1091-5x & 1015-54 If no connector-name is given, then connector will be set to this. Query response string: “TNCF”

COAX <connector>	[connector-name]	Description
SMAMale	3650	SMA(M) Cal Kit: 3650 If no connector-name is given, then connector will be set to this. Query response string: “SMAM”
SMAFemale	3650	SMA(F) Cal Kit: 3650 If no connector-name is given, then connector will be set to this. Query response string: “SMAF”
431Female	2000-1914-R	COAX DUT 4.3-10 Female
431Male	2000-1915-R	COAX DUT 4.3-10 Male
USR1		Query response string: “USR1”
USR2		Query response string: “USR2”
USR3		Query response string: “USR3”
USR4		Query response string: “USR4”

Syntax:
```
[:SENSe]:CORRection:COLLect:CONNector<port_no>
<connector>,  [connector-name string]
[:SENSe]:CORRection:COLLect:CONNector<port_no>?
```

Cmd Parameter:
```
<char> <connector>,  [connector-name string]
```

Query Response:
```
<char> <connector><(connector-name string)>
(connector is returned in short form only)
```

Default Value: NMAL

Example: To set the DUT connector for port 1 to K-Conn(M) Cal Kit: OSLK50.

```
:SENS:CORR:COLL:CONN1 KMAL, "OSLK50"
```

or

```
:SENS:CORR:COLL:CONN1 KMAL
```

To query for the DUT connector for port 1:

```
:SENS:CORR:COLL:CONN1?
```

For the example above the query response is: KMAL

To set the DUT connector for port 1 to K-Conn(M)
Cal Kit: TOSLK50A-20

```
:SENS:CORR:COLL:CONN1 KMAL, "TOSLK50A"
```

or

```
:SENS:CORR:COLL:CONN1 KMAL, "TOSLK50A-20"
```

The query response would return the following:

```
KMAL(TOSLK50A-20)
```

Front Panel Access: **Shift-2** (Calibrate), DUT Port Setup, DUT Port 1

Calibration Interpolation

[:SENSe]:CORRection:COLLect:INTerpolation[:STATe]

Description: Turns the calibration interpolation ON or OFF. Note that interpolation automatically turns On after a flex cal.

Syntax:
```
[:SENSe]:CORRection:COLLect:INTerpolation[:STATe] OFF|ON
[:SENSe]:CORRection:COLLect:INTerpolation[:STATe]?
```

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: 0

Example: To set interpolation to ON:

```
:SENS:CORR:COLLect:INT:STAT ON
or
:CORR:COLLect:INT 1
```

Front Panel Access: **Shift-2** (Calibrate), Interpolation

Calibration Line Type

[:SENSe]:CORRection:COLLect:MEDium

Description: Sets the calibration line type.

Syntax:
```
[:SENSe]:CORRection:COLLect:MEDium COAX
[:SENSe]:CORRection:COLLect:MEDium?
```

Cmd Parameter: `<char> COAX`

Query Response: `<char> COAX`

Default Value: COAX

Front Panel Access: **Shift-2** (Calibrate)

Calibration Method

[:SENSe]:CORRection:COLLect:METHod

Description: Sets the calibration method.

Syntax: `[:SENSe]:CORRection:COLLect:METHod SOLT`
`[:SENSe]:CORRection:COLLect:METHod?`

Parameter: `SOLT`

Cmd Parameter: `<char> SOLT`

Query Response: `<char> SOLT`

Default Value: SOLT

Front Panel Access: **Shift-2** (Calibrate)

Calculate Calibration Data

[:SENSe]:CORRection:COLLect:SAVe

Description: Calculates the calibration data according to the calibration method that is selected and the steps that are performed and then stores the result. This command is similar to the "calculate and finish" on the front panel of the list of calibration steps.

Syntax: `[:SENSe]:CORRection:COLLect:SAVe`

Cmd Parameter: **NA**

Query Response: **NA** (no query)

Front Panel Access: **NA**

Calibration Status

[:SENSe]:CORRection:COLLect:STATus?

Description: This command requests information about the calibration status. The command returns `0` if none, `1` if calibration has already started, `2` if calibration has been aborted, `3` if a calibration is currently calculating, and `4` if a calibration has been completed.

Syntax: `[:SENSe]:CORRection:COLLect:STATus?`

Cmd Parameter: **NA** (query only)

Query Response: `<char> [<cal steps>, <port_no>]`

Front Panel Access: **NA**

Calibration Accuracy Status

[:SENSe]:CORRection:COLLect:STATus:ACCuracy?

Description: This command requests information about the calibration accuracy status. The command returns:

- `0` when no calibration is available (CAL off)
- `1` when the calibration accuracy is high (OK: Accuracy High)
- `2` when the accuracy is moderate due to a change in power level (?P: Accuracy Moderate)
- `3` when the accuracy is moderate due to a change in temperature level by more than 5 deg C (?T: Accuracy Moderate)
- `4` when the accuracy is low due to a change in temperature level by more than 10 deg C (X: Accuracy Low).

Syntax: `[:SENSe]:CORRection:COLLect:STATus:ACCuracy?`

Cmd Parameter: **NA** (query only)

Query Response: `<NR1> (integer)`

```
0 = CAL off
1 = OK: Accuracy High
2 = ?P: Accuracy Moderate
3 = ?T: Accuracy Moderate
4 = X: Accuracy Low
```

Front Panel Access: **Shift 2** (Calibrate), Existing Cal Info

Calibration Type

[:SENSe]:CORRection:COLLect:TYPE

Description: Configures the calibrate type. `<cal type>` is the calibration type and must be one of the following values:

`RFP1|TRFP|RRP1|2PFP`

CAL TYPE	DESCRIPTION
RFP1	Full Port 1
TRFP	Trans Response Fwd Path
RRP1	Reflection Response Port 1
2PFP	1 Path 2 Port Fwd Path

Syntax:
```
[:SENSe]:CORRection:COLLect:TYPE <cal type>
[:SENSe]:CORRection:COLLect:TYPE?
```

Cmd Parameter: `<char> <cal type>`

Query Response: `<char> <cal type>`

Default Value: 2PFP

Front Panel Access: **Shift-2** (Calibrate), Cal Type

3-24 [:SENSe]:DITHering

Turns the Dithering function on or off. When a signal is dithered, a mathematical process removes the harmonics or other undesirable distortions, and replaces these distortions with a constant fixed noise level. The result is a better representation of the signal.

[:SENSe]:DITHering:[STATe]OFF|ON|0|1

Description: Turns Dithering on or off.

Syntax: `[:SENSe]:DITHering:[STATE]OFF|ON|0|1`

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1>0|1`

Default Value: OFF

3-25 [:SENSe]:FREQuency Subsystem

Commands in this subsystem pertain to the frequency settings of the instrument.

Center Frequency

[:SENSe]:FREQuency:CENTer

Description: Sets the center frequency. Note that changing the value of the center frequency will change the value of the coupled parameters: `Start Frequency` and `Stop Frequency`. It may also change the value of the span.

Syntax:
```
[:SENSe]:FREQuency:CENTer <freq>
[:SENSe]:FREQuency:CENTer?
```

Cmd Parameter: `<NRf> <freq> (hertz)`

Query Response: `<NR3> <freq> (hertz)`

Default Value: 2000250000 Hz for MS2024B, MS2034B
3000250000 Hz for MS2025B, MS2035B

Default Units: Hz

Range: 500 kHz to 4 GHz for MS2024B, MS2034B
500 kHz to 6 GHz for MS2025B, MS2035B

Front Panel Access: **Freq/Dist** (or **Freq**), Center Freq

Frequency Span

[:SENSe]:FREQuency:SPAN

Description: Sets the frequency span. Setting the value of `<freq>` to 0 Hz is the equivalent of setting the span mode to zero span. Note that changing the value of the frequency span will change the value of coupled parameters: Start Frequency and Stop Frequency, and may change the Center Frequency.

Syntax:
```
[:SENSe]:FREQuency:SPAN <freq>
[:SENSe]:FREQuency:SPAN?
```

Cmd Parameter: `<NRf> <freq> (hertz)`

Query Response: `<NR3> <freq> (hertz)`

Default Value: 3999500000 Hz for MS2024B, MS2034B
5999500000 Hz for MS2025B, MS2035B

Default Units: Hz

Range: 0 Hz to 3.999500 GHz for MS2024B, MS2034B
0 Hz to 5.999500 GHz for MS2025B, MS2035B

Front Panel Access: **Freq/Dist** (or **Freq**), Span

Distance Suggested Frequency Span

[:SENSe]:FREQuency:DSPAn?

Description: This command returns the suggested frequency span based on the start and stop distance.

Syntax: `[:SENSe]:FREQuency:DSPAn?`

Cmd Parameter: **NA** (query only)

Query Response: `<NR3> <freq> (hertz)`

Default Value: At preset, DSpan is 1999750000 Hz for MS2024B, MS2034B
At preset, DSpan is 2999750000 Hz for MS2025B, MS2035B

Default Units: Hz

Range: 0 Hz to 1.999750000 GHz for MS2024B, MS2034B
0 Hz to 2.999750000 GHz for MS2025B, MS2035B

Front Panel Access: **Freq/Dist**, Additional Dist Setup, Distance Info

Start Frequency

[:SENSe]:FREQuency:STARt

Description: Sets the start frequency. Note that changing the value of the start frequency will also change the value of coupled parameters: `Center Frequency` and `Span`.

Syntax:
```
[:SENSe]:FREQuency:STARt <freq>
[:SENSe]:FREQuency:STARt?
```

Cmd Parameter: `<NRf> <freq> (hertz)`

Query Response: `<NR3> <freq> (hertz)`

Default Value: 500000 Hz for all 4 models

Default Units: Hz

Range: 500 kHz to 4 GHz for MS2024B, MS2034B
500 kHz to 6 GHz for MS2025B, MS2035B

Example: Sets the start frequency to 10000 HZ:

```
:SENSe:FREQuency:STARt 10000
```

Sets the start frequency to 5 MHz:

```
:SENSe:FREQuency:STARt 5MHZ
```

Sets the start frequency to 1 GHz:

```
:SENS:FREQ:STAR 1GHZ
```

Front Panel Access: **Freq/Dist** (or **Freq**), Start Freq

Stop Frequency

[:SENSe]:FREQuency:STOP

Description: Sets the stop frequency. Note that changing the value of the stop frequency will change the value of coupled parameters: `Center Frequency` and `Span`.

Syntax:
```
[:SENSe]:FREQuency:STOP <freq>
[:SENSe]:FREQuency:STOP?
```

Cmd Parameter: `<NRf> <freq> (hertz)`

Query Response: `<NR3> <freq> (hertz)`

Default Value: 4000000000 Hz for MS2024B, MS2034B
6000000000 Hz for MS2025B, MS2035B

Default Units: Hz

Range: 500 kHz to 4 GHz for MS2024B, MS2034B
500 kHz to 6 GHz for MS2025B, MS2035B

Example: Sets the stop frequency to 10000 Hz:

```
:SENSe:FREQuency:STOP 10000
```

Sets the stop frequency to 5 MHz:

```
:SENSe:FREQuency:STOP 5MHZ
```

Sets the stop frequency to 1 GHz:

```
:SENS:FREQ:STOP 1GHZ
```

Front Panel Access: **Freq/Dist** (or **Freq**), Stop Freq

Get Frequency List

:SENSe<Tr>:FREQuency:DATA?

Description: Produces the frequency list in Hz for the given trace. <Tr> is the trace number in the range 1 to 8 (1 to 4 for Traces TR1 to TR4 and 5 to 8 for Memory M1 to M4). If no trace number is specified, then the <Tr> parameter defaults to trace number 1. The response begins with an ASCII header that specifies the number of data bytes. It appears in the format #AX, where A is the number of digits in X, and X is the number of bytes that follow the header. Each frequency point is in scientific notation and separated by a comma delimiter.

Syntax: :SENSe<Tr>:FREQuency:DATA?

Cmd Parameter: **NA** (query only)

Query Response: <char> <freq> (returns block data in hertz)

Default Units: Hz

Related Command: :CALCulate<Tr>:TRANsform:DISTance:DATA?

Front Panel Access: **NA**

3-26 [:SENSe]:SWEep Subsystem

This subsystem includes commands that affect the sweep parameters of the instrument.

IF Bandwidth

[:SENSe]:SWEep:IFBW

Description: Sets the IF Bandwidth. When the Measurement View is set to VNA, the `<freq value>` in Hz must be one of the following 13 values:

`100000|50000|20000|10000|5000|2000|1000|500|200|100|50|20|10`

When the Measurement view is set to FIELD, the `<freq value>` in Hz must be one of the following 3 values:

`100000|1000|10`

The query form of this command returns the frequency in Hz.

Syntax:
```
[:SENSe]:SWEep:IFBW <freq value>
[:SENSe]:SWEep:IFBW?
```

Cmd Parameter: `<char> <freq value>`

Query Response: `<char> <freq value>`

Default Value: 10000

Default Units: Hz

Range: 100 to 100000 Hz

Example: Sets the IF Bandwidth frequency to 200 Hz:

```
:SENS:SWE:IFBW 200
```

Sets the IF Bandwidth frequency to 100 kHz:

```
:SENS:SWE:IFBW 100000
```

Front Panel Access: **Shift-3** (Sweep), IFBW

Number of Sweep Points

[:SENSe]:SWEep:POINts

Description: Sets the total number of measurement points per sweep. Note that a sweep with a lower number of data points will complete in less time than a sweep with a higher number of data points.

Syntax:
```
[:SENSe]:SWEep:POINts <integer>
[:SENSe]:SWEep:POINts?
```

Cmd Parameter: `<NR1> <integer>`

Query Response: `<NR1> <integer>`

Default Value: 201

Range: 2 to 4001

Front Panel Access: **Shift-3** (Sweep), Data Points

Sweep Type

[:SENSe]:SWEep:TYPE

Description: Sets the sweep type. The query version of this command returns "SING" if current sweep is set to single sweep, and returns "CONT" if set to continuous sweep. Note that setting the sweep type to SINGle sets the sweep to hold.

Syntax:
```
[:SENSe]:SWEep:TYPE SINGle|CONTinuous
[:SENSe]:SWEep:TYPE?
```

Cmd Parameter: `<char> SINGle|CONTinuous`

Query Response: `<char> SING|CONT`

Default Value: CONT

Front Panel Access: **Shift-3** (Sweep), Sweep Type

3-27 [:SENSe]:TRACe Subsystem

This subsystem includes commands that provide general settings for each trace.

Trace Domain

[:SENSe]:TRACe<Tr>:DOMain

Description: Defines the domain for the given trace <Tr>. <Tr> is the trace number in the range 1 to 4. If no trace number is specified, then the <Tr> parameter defaults to trace number 1. The query version of this command returns "FREQ" if domain is Frequency, and "DIST" if domain is distance.

Syntax:
```
[:SENSe]:TRACe<Tr>:DOMain FREQuency|DISTance
[:SENSe]:TRACe<Tr>:DOMain?
```

Cmd Parameter: <char> FREQuency|DISTance

Query Response: <char> FREQ|DIST

Default Value: Trace 1: FREQ
Trace 2: FREQ
Trace 3: FREQ
Trace 4: FREQ

Front Panel Access: **Measure**, Domain Selection

Trace Select

[:SENSe]:TRACe<Tr>:SELect

Description: Selects the given trace, <Tr>, as the active trace. <Tr> is the trace number in the range 1 to 4. If no trace number is specified, then the <Tr> parameter defaults to trace number 1. Note that this may also change the total number of traces.

Syntax: [:SENSe]:TRACe<Tr>:SELect

Cmd Parameter: **NA**

Query Response: **NA** (no query)

Default Value: TR1

Example: To set trace 2 as the active trace:

```
:SENSe:TRACe2:SELect
:SENS:TRAC2:SEL
```

To set trace 1 as the active trace:

```
:SENSe:TRACe1:SELect
:SENSe:TRACe:SELect
```

Front Panel Access: **Measure**, Active Trace

S Parameter

[:SENSe]:TRACe<Tr>:SPARams

Description: Defines the S-parameter for the given trace, `<Tr>`.

`<Tr>` is the trace number in the range 1 to 4. If no trace number is specified, then the `<Tr>` parameter defaults to trace number 1. The query version of this command returns "`S11`" if the S-parameter is set to S11, "`S21`" if set to S21.

Syntax:
```
[:SENSe]:TRACe<Tr>:SPARams S11|S21
[:SENSe]:TRACe<Tr>:SPARams?
```

Cmd Parameter: `<char> [S11|S21`

Query Response: `<char> [S11|S21`

Default Value:
Trace 1: S11
Trace 2: S21
Trace 3: S11
Trace 4: S21

Example: To assign S11 to trace 2:

```
:SENSe:TRACe2:SPARams S11
```

Front Panel Access: **Measure**, S Parameter

Number of Traces

[:SENSe]:TRACe:TOTal

Description: Sets the number of traces to display.

Syntax:
```
[:SENSe]:TRACe:TOTal <integer>
[:SENSe]:TRACe:TOTal?
```

Cmd Parameter: `<char> [1|2|3|4]`

Query Response: `<char> [1|2|3|4]`

Default Value: 2

Range: 1 to 4

Example: To set number of traces to 3:

```
:SENSe:TRACe:TOTal 3
```

Front Panel Access: **Measure**, Number of Traces

Active Trace

[:SENSe]:TRACe:SELect?

Description: This command returns the current active trace number in the format `TR#`.

Syntax: `[:SENSe]:TRACe:SELect?`

Cmd Parameter: **NA** (query only)

Query Response: `<char> [TR1|TR2|TR3|TR4]`

Example: To query for the active trace number:

`:SENS:TRAC:SEL?`

Front Panel Access: **Measure**, Active Trace

3-28 :SOURce Subsystem

The commands in this subsystem control the internal signal source.

Table 3-9. :SOURce Subsystem

Keyword	Parameter Data or Units
`:SOURce`	
`:CORRection`	Refer to ":SOURce:CORRection:RVELocity Subsystem" on page 3-101

Power Levels

`:SOURce:POWer`

Description: Sets the power levels.

Syntax:
```
:SOURce:POWer LOW|HIGH|DFLT
:SOURce:POWer?
```

Cmd Parameter: `<char> [LOW|HIGH|DFLT]`

Query Response: `<char> [LOW|HIGH|DFLT]`

Default Value: DFLT

Range: HIGH: +6 dBm to 0 dBm
DFLT: 0 dBm to –6 dBm
LOW: –15 dBm to –25 dBm

Front Panel Access: **Shift-3** (Sweep), Configure Ports, Source Power

3-29 :SOURce:CORRection:RVELocity Subsystem

Commands in this subsystem deal with the parameters of the physical media of the Device Under Test.

Propagation Velocity

:SOURce:CORRection:RVELocity

Description: Sets the propagation velocity of the cable for DTF measurements.

Syntax:
```
:SOURce:CORRection:RVELocity <number>
:SOURce:CORRection:RVELocity?
```

Cmd Parameter: `<NRf> <number> (unitless)`

Query Response: `<NR3> <number> (unitless)`

Default Value: 1

Range: 0.001 to 1.0

Front Panel Access: **Freq/Dist** (or **Freq**), Domain Setup, Setup Distance, Additional Dist Setup, Propagation Velocity

Cable Loss

:SOURce:CORRection:RVELocity:CABLoss

Description: Sets the cable loss for DTF measurements. The query version of this command returns the cable loss in dB/m.

Syntax:
```
:SOURce:CORRection:RVELocity:CABLoss <number>
:SOURce:CORRection:RVELocity:CABLoss?
```

Cmd Parameter: `<NRf> <number> (unitless)`

Query Response: `<NR3> <number> (unitless)`

Default Value: 0.0

Range: 0.0 to 5

Front Panel Access: **Freq/Dist** (or **Freq**), Domain Setup, Setup Distance, Additional Dist Setup, Cable Loss (when DUT Line Type is Coax)

3-30 :STATus Subsystem

The commands in this subsystem relate to the current operating state of the instrument.

Query Operation Status

`:STATus:OPERation?`

Description: This command requests information about the current status of the instrument. Each bit of the return value represents some operation. Only a subset of the bits are implemented for each application. The number returned is the decimal representation of the bit-wise `OR` of the enabled bits.

Bit	Decimal Value	Description
0	1	Not implemented
1	2	Not implemented
2	4	Not implemented
3	8	Not implemented
4	16	Not implemented
5	32	Not implemented
6	64	Not implemented
7	128	Not implemented
8	256	Sweep Complete – This bit will be set to 0 when the command :INITiate[:IMMediate] is sent to trigger a sweep. It will have a value of 1 when the sweep has completed.
9	512	Not implemented
10	1024	Not implemented
11	2048	Not implemented
12	4096	Not implemented
13	8192	Not implemented
14	16384	Not implemented
15	0	Will always be 0

Syntax: `:STATus:OPERation?`

Cmd Parameter: **NA** (query only)

Query Response: `<decimal> (0 to 15 bit)`

Front Panel Access: **NA**

3-31 :SYSTem Subsystem

The commands in this subsystem relate to the current operating state of the instrument.

System Alternate State

:SYSTem:ALTernate:STATe

Description: Sets the state of the Measurement View to be either VNA or Field. The query form of this command returns a 0 for VNA and a 1 for Field.

Syntax:
```
:SYSTem:ALTernate:STATe VNA|FIELD
:SYSTem:ALTernate:STATe?
```

Cmd Parameter: `<char> VNA|FIELD`

Query Response: `char 0|1`

Default Value: VNA

Example: To set the Measurement View to Field:

```
:SYSTem:ALTernate:STATe FIELD
```

To set the Measurement View to VNA:

```
:SYSTem:ALTernate:STATe VNA
```

Front Panel Access: **Shift-8 (System)**, Application Option, Meas Menu

Motherboard Temperature

:SYSTem:MBTemperature?

Description: This command returns the current mother board temperature in degrees Celsius.

Syntax: `:SYSTem:MBTemperature?`

Cmd Parameter: **NA** (query only)

Query Response: `<NR3> (degree Celsius)`

Front Panel Access: **NA**

Preset

:SYSTem:PRESet

Description: This command restores all application parameters to their factory preset values. This command does not modify system parameters such as Ethernet configuration, language, volume, or brightness.
Note: After issuing this command, wait 30 seconds for the instrument to complete the preset condition before sending another command.

Syntax: `:SYSTem:PRESet`

Cmd Parameter: **NA**

Query Response: **NA** (no query)

Related Command: `*RST`

Front Panel Access: **Shift-1** (Preset), Preset

3-32 :Trace Subsystem

This subsystem contains commands related to the transfer of trace data to and from the instrument.

Trace Data Transfer

:TRACe[:DATA]?

Description: Transfers the trace data of the given trace from the instrument to the controller.

The format of the block data that is returned can be specified by the command `:FORMat:DATA`. The response begins with an ASCII header that specifies the number of data bytes. It appears in the format `#AX`, where `A` is the number of digits in `X`, and `X` is the number of bytes that follow the header. Each data point is separated by a comma delimiter. Independent of the Graph Type that is associated with the trace, each data point that is transferred by this command consists of complex measurement data (Real and Imaginary values for that point). A 551 point trace therefore has a total of 1102 points that get transferred.

Syntax: `:TRACe[:DATA]? [1]|2|3|4`

Cmd Parameter: **NA** (query only)

Query Response: `<block> (returns block data)`

Related Command: `:FORMat:DATA;`
`:CALCulate<Tr>:DATA?`

Front Panel Access: **NA**

Trace Header Transfer

:TRACe:PREamble?

Description: Returns trace header information for the specified trace. Data can be transferred from the 4 available display traces. Use the commands in the **MMEMory** subsystem to store and recall traces from the instrument memory. The response begins with an ASCII header. The header specifies the number of following bytes. It appears in the format `#AX`, where `A` is the number of digits in `X`, and `X` is the number of bytes that follow the header. Refer to the following section, "Example Response Format:".

Parameters are returned in comma-delimited ASCII format. Each parameter is returned as `"NAME=VALUE[UNITS]"`. Note that the parameters that are returned depend on the firmware version and that this document does not cover all parameter values that are returned by the command.

Syntax: `:TRACe:PREamble? [1]|2|3|4`

Cmd Parameter: **NA** (query only)

Query Response: `<block> (returns block data)`

Front Panel Access: **NA**

Example Response Format:

#AX is `#40078`, where `A = 4` (the number of digits in number X), and `X = 0078` (the response has 78 characters).

#40078`SN=6897458,TYPE=DATA,DATE=2009-03-18-03-13-20-00,INT_BIAS_TEE_CURRENT=0.000000`

Parameters are returned in comma-delimited ASCII format. Each parameter is returned as "NAME=VALUE[UNITS]". For the example response, the serial number (SN) is 6897458 and is returned as `"SN=6897458"`.

Note	The parameters that are returned depend on the firmware version in the MS20xxB, and this document does not cover all possible parameter values that can be returned by the command.

The following 3 tables describe some of the common parameters that can be returned by the `:TRACe:PREamble?` command:

- Table 3-10, "Trace Header Parameters" on page 3-107
- Table 3-11, "Trace Header Marker Parameters" on page 3-115
- Table 3-12, "Trace Header Limits Parameters" on page 3-116

Trace Header Transfer (continued)

Table 3-10. Trace Header Parameters (Sheet 1 of 8)

Parameter Name	Description
SN	Instrument serial #
UNIT_NAME	Instrument name
TYPE	The data type (Setup or Data)
DATE	Trace date/time
APP_NAME	Application name
APP_VER	Application firmware (FW) version
SUB_MODE	Sub Mode type, where: 0 is for Vector Network Analyzer, 2 is for Vector Voltmeter
S_TYPE	Active trace S type. Current available S Types are: S11 = 0, S21 = 1
TRACE_S_TYPES	S types for all 4 traces. This uses a bit mask, where the bit shift mask is defined as: S_TYPE_BIT_SHIFT 4 S_TYPE_BIT_MASK 0xF For example, to get the S type for trace 1: (int) (sTypes >> (S_TYPE_BIT_SHIFT * 0)) & S_TYPE_BIT_MASK
GRAPH_TYPE	Active Trace graph type. Current available graph types are: Log Mag = 0, SWR = 1, Phase = 2, Real = 3, Imaginary = 4, Group delay = 5, Smith Chart = 6, Log Mag/2 = 7 Linear Polar = 8 Log Polar = 9 Real Impedance = 10 Imaginary Impedance = 11
TRACE_GRAPH_TYPES	Graph types for all 4 traces. This uses a bit mask shift, where the bit shift mask is defined as: GRAPH_TYPE_BIT_SHIFT 16 GRAPH_TYPE_BIT_MASK 0xFFFF

Table 3-10. Trace Header Parameters (Sheet 2 of 8)

Parameter Name	Description
DOMAIN	Active Trace domain type, where: 0 is frequency domain, 2 is for distance domain
TRACE_DOMAIN_TYPES	Domain types for all 4 traces. This uses a bit mask shift, where the bit shift mask is defined as: DOMAIN_TYPE_BIT_SHIFT 4 DOMAIN_TYPE_BIT_MASK 0xF
DOMAIN_SETUP	Current Domain Setup. Available Domain setups are: Freq = 0, Dist = 2
TRACE_MATH_TYPES	Trace Math types. This uses a bit mask shift, where the bit shift mask is defined as: MATH_TYPE_BIT_SHIFT 4 MATH_TYPE_BIT_MASK 0xF Available math type are: None = 0, Subtract = 1, Add = 2, Multiply = 3, Divide = 4
TRACE_DISPLAY_TYPES	Trace display types. Available trace types are: Trace Only = 0, Memory Only = 1, Trace and Memory = 2
TRACE_MEMORY_STATE	For save/recall purpose. Where 0 is Off and 1 is On.
SMITH_CHART_TYPE	Current active trace Smith Chart type. Available Smith Chart type are: Normal = 0, Expand 10dB = 1, Expand 20dB = 2, Expand 30dB = 3, Compress 3dB = 4
TRACE_SMITH_CHART_TYPES	Smith Chart type. This uses a bit mask shift, where the bit shift mask is defined as: SMITH_CHART_TYPE_BIT_SHIFT 4 SMITH_CHART_TYPE_BIT_MASK 0xF For available types, refer to “SMITH_CHART_TYPE”.
SMITH_REF_IMPED	Reference Impedance. Where: 50 ohm = 0. and 75 ohm = 1
TOTAL_CHANNELS	Trace Format. Available trace format are: Single = 1, Dual = 2, Tri = 3, Quad = 4

Table 3-10. Trace Header Parameters (Sheet 3 of 8)

Parameter Name	Description
ACTIVE_TRACE	Current active trace. Where: 0 is for trace 1, 1 for trace 2, 2 for trace 3, and 3 for trace 4
TOTAL_TRACE	Total number of traces
AVERAGING_COUNT	Current Averaging Count
AVERAGING_FACTOR	Averaging factor
EXTERNAL_REFERENCE	External Reference where 0 is for Off and 1 is for Locked
EXT_REF_FREQ_LIST	Currently not being used.
SWEEP_TYPE	Sweep type. Available sweep types are: Single = 0 and Continuous = 1
EXTERNAL_TRIGGER	Currently not being used.
BIAS_TEE_STATE	Bias Tee State. Currently available Bias Tee states are: Off = 0 and Internal = 2
BIAS_TEE_PORT_SELECTION	Bias Tee port selection. Where: 1 is port 2.
BIAS_TEE_VOLTAGE_Px	Internal Bias Tee voltage, where: x is the port number. Return value is 1000 times the current voltage value in Volts.
INT_BIAS_TEE_VOLTAGE	Internal Bias Tee voltage. Return value in mV. Internal Bias Tee current limit, where: x is the port number. Return value is in mA.
BIAS_TEE_CURRENT_LIMIT_Px	Internal Bias Tee current limit, where: x is the port number. Return value is in mA.
INT_BIAS_TEE_CURRENT	Internal Bias Tee current. Return value is in mA.
RF_SOURCE_POWER	Source Power. Current valid source power: low = 0, default = 1, and high = 2.
CABLE	The index of the selected cable list, where 0 is the first in the list.
DIST_UNITS	Distance units. Available distance units are: Meter = 0, Feet = 1

Table 3-10. Trace Header Parameters (Sheet 4 of 8)

Parameter Name	Description
IFBW	The index of the selected IFBW list, where 0 is the first in the list.
DUT_LINE_TYPE	DUT Line Type, where Coax = 0.
CUTOFF_FREQ	Currently not used.
PROP_VEL	Propagation Velocity. Value returned is 1000 times the propagation velocity value.
CABLE_LOSS	Cable Loss. Value returned is 1000 times the cable loss value.
MARKER_SELECTED	The current selected marker, where marker # is the value + 1. For example, a value of 0 is marker number 1.
MARKER_TYPE	The current selected marker type. Where: Ref = 0 delta = 1 off = 2
MARKER_TABLE	Currently not being used.
MARKER_READOUT_STYLE	The current selected marker readout style. Available readout styles are: Graph = 0, Log Mag = 1 Log Mag and Phase = 2 Phase = 3 Real and Imaginary = 4 SWR = 5 Impedance = 6 Admittance = 7 Normalized Impedance = 8 Normalized Admittance = 9 Polar Impedance = I0 Group Delay = 11 Log Mag/2 = 12 Lin Mag = 13 Lin Mag and Phase = 14
MARKER_READOUT_FORMAT	Marker Readout Format. Available readout formats are: None = 0 Trace =1 Screen = 2 Table = 3
PORT_x_REF_PLANE_LENGTH	Reference Plane Length, where x is the port number. Returns in units of meter.

Table 3-10. Trace Header Parameters (Sheet 5 of 8)

Parameter Name	Description
TRACE_SMOOTHING_PERCENT	Trace smoothing percent. This uses a bit mask shift, where the bit shift mask is defined as: PERC_SMOOTHING_BIT_SHIFT 8 PERC_SMOOTHING_BIT_MASK 0xFF
SMOOTHING_PERCENT	Current active trace smoothing percent.
CURRENT_LIMIT	The limit type (upper/lower) for the active trace. Upper = 0 and Lower = 1
LIMIT_STATE	The limit state (on/off) for the active trace. On = 0, Off = 1
LIMIT_ALARM	The limit alarm (on/off) for the active trace. On = 0, Off = 1
LIMIT_MESSAGE	Limit Pass Fail Message (on/off) for the active trace. On = 0, Off = 1.
CURRENT_TEMPERATURE	The current temperature. Valid only with a cal. To get the temperature in Celsius divide the result by 4.
TRACE_x_LP_MODE	Currently not used.
TRACE_x_LP_RESPONSE_TYPE	Currently not used.
TRACE_x_LP_PHASOR_IMPULSE	Currently not used.
TRACE_x_POLAR_RESOLUTION	Linear Polar Resolution, where x is the trace number. Returned value is 1000 times the resolution value
TRACE_x_POLAR_REFERENCE	Linear Polar Reference value, where x is the trace number. Returned value is 1000 times the reference value
TRACE_x_POLAR_REFERENCE_LINE	Currently not used
TRACE_x_LOG_POLAR_RESOLUTION	Log Polar Resolution, where x is the trace number. Returned value is in dB.
TRACE_x_LOG_POLAR_REFERENCE	Log Polar Reference value, where x is the trace number. Returned value is in dB.
TRACE_x_LOG_POLAR_REFERENCE_LINE	Currently not used
TRACE_x_REAL_Z_RESOLUTION	Real Impedance resolution, where x is the trace number. Returned value is 1000 times the resolution value.
TRACE_x_REAL_Z_REFERENCE	Real Impedance Reference value, where x is the trace number. Returned value is 1000 times the reference value.
TRACE_x_REAL_Z_REFERENCE_LINE	Real Impedance Reference line, where x is the trace number.

Table 3-10. Trace Header Parameters (Sheet 6 of 8)

Parameter Name	Description
TRACE_x_IMAG_Z_RESOLUTION	Imaginary Impedance resolution, where x is the trace number. Returned value is 1000 times the resolution value.
TRACE_x_IMAG_Z_REFERENCE	Imaginary Impedance Reference value, where x is the trace number. Returned value is 1000 times the reference value.
TRACE_x_IMAG_Z_REFERENCE_LINE	Imaginary Impedance Refernece line, where x is the trace number.
TRACE_x_START_FREQ	Start freq, where x is the trace number. Returns in units of megahertz.
TRACE_x_STOP_FREQ	Stop frequency, where x is the trace number. Returns in units of megahertz.
TRACE_x_CENTER_FREQ	Center frequency, where x is the trace number. Returns in units of megahertz.
TRACE_x_SPAN	Frequency span, where x is the trace number. Returns in units of megahertz.
TRACE_x_START_DIST	Start distance, where x is the trace number. Depending on the given distance unit, value is returned in units of either microfeet or micrometer.
TRACE_x_STOP_DIST	Stop distance, where x is the trace number. Depending on the given distance unit, value is returned in units of either microfeet or micrometer.
TRACE_x_SMOOTHING_PERCENT	Currently not used.
TRACE_x_WINDOWING	Windowing, where x is the trace number. Available windowing settings are: Rectangular = 0, Nominal Side Lobe = 1, Low Side Lobe = 2, Minimum Side Lobe = 3
TRACE_x_GD_APERTURE	Group Delay Aperture, where x is the trace number.
TRACE_x_DSP_DATA_POINTS	Number of data points, where x is the trace number.
TRACE_x_LOG_MAG_RESOLUTION	Log Mag Resolution, where x is the trace number. Returned value is in dB.
TRACE_x_LOG_MAG_REFERENCE	Log Mag Reference value, where x is the trace number. Returned value is in dB.
TRACE_x_LOG_MAG_REFERENCE_LINE	Log Mag Reference Line, where x is the trace number.

Table 3-10. Trace Header Parameters (Sheet 7 of 8)

Parameter Name	Description
TRACE_x_SWR_RESOLUTION	SWR Resolution, where x is the trace number. Returned value is 1000 times the SWR Resolution.
TRACE_x_SWR_REFERENCE	SWR Reference value, where x is the trace number. Returned value is 1000 times the SWR Reference value.
TRACE_x_SWR_REFERENCE_LINE	SWR Reference Line, where x is the trace number.
TRACE_x_PHASE_RESOLUTION	Phase Resolution, where x is the trace number. Returned value is 1000 times the phase resolution.
TRACE_x_PHASE_REFERENCE	Phase Reference value, where x is the trace number. Returned value is 1000 times the phase reference value.
TRACE_x_PHASE_REFERENCE_LINE	Phase Reference Line, where x is the trace number.
TRACE_x_REAL_RESOLUTION	Real Resolution, where x is the trace number. Returned value is 1000 times the resolution per div value.
TRACE_x_REAL_REFERENCE	Real Reference value, where x is the trace number. Returned value is 1000 times the reference value.
TRACE_x_REAL_REFERENCE_LINE	Real Reference line, where x is the trace number.
TRACE_x_IMAG_RESOLUTION	Imaginary Resolution, where x is the trace number. Returned value is 1000 times the reference value.
TRACE_x_IMAG_REFERENCE	Imaginary Reference value, where x is the trace number. Returned value is 1000 times the reference value.
TRACE_x_IMAG_REFERENCE_LINE	Imaginary Reference line, where x is the trace number.
TRACE_x_GD_RESOLUTION	Group Delay Resolution, where x is the trace number. Returns in units of picoseconds.
TRACE_x_GD_REFERENCE	Group Delay Reference value, where x is the trace number. Returns in units of picoseconds.
TRACE_x_GD_REFERENCE_LINE	Group Delay Reference line, where x is the trace number.
TRACE_x_SMITH_SCALE	Currently not used.
TRACE_x_SMITH_IMPEDANCE	Currently not used.
TRACE_x_SMITH_IMPEDANCE_LINE	Currently not used.

Table 3-10. Trace Header Parameters (Sheet 8 of 8)

Parameter Name	Description
TRACE_x_1PCL_RESOLUTION	Log Mag/2 resolution, where x is the trace number. Returned value is in dB.
TRACE_x_1PCL_REFERENCE	Log Mag/2 Reference value, where x is the trace number. Returned value is in dB.
TRACE_x_1PCL_REFERENCE_LINE	Log Mag/2 Reference line, where x is the trace number.
TRACE_x_POLAR_RESOLUTION	Currently not used.
TRACE_x_POLAR_REFERENCE	Currently not used.
TRACE_x_POLAR_REFERENCE_LINE	Currently not used.
CAL_METHOD	Calibration Method, where: SOLT = 0
CAL_TYPE	The index of the selected calibration type list, where 0 is the first in the list.
CAL_LINE_TYPE	Calibration Line Type, where coax = 0.
CAL_PORTx_DUT	The index of the selected Calibration Coax DUT Selector list for port x, where 0 is the first in the list.
CAL_CORRECTION	Calibration correction, where On = 0, and Off = 1
APP_SELF_TEST_MODE	Internal use only
DEBUG_MEAS_GAIN_RANGE	Internal use only
LOG_COUNTER_EVENTS	Internal use only
SWEEP_DEFAULT_FREQS	Internal use only
PWRCAL_RF_SWITCH_FREQ	Internal use only
PWRCAL_LOW_RF_HIGH_TARGET	Internal use only
PWRCAL_LOW_RF_LOW_TARGET	Internal use only
PWRCAL_UPPER_RF_HIGH_TARGET	Internal use only
PWRCAL_UPPER_RF_LOW_TARGET	Internal use only
PWRCAL_UW_RF_HIGH_TARGET	Internal use only
PWRCAL_UW_RF_LOW_TARGET	Internal use only
USER_DEFINED_CAL_KIT_NAME	Internal use only
USER_DEFINED_CAL_KIT	Internal use only
TRACE_LABEL_STATE	Trace label On/Off, where On = 0, Off = 1

Trace Header Transfer (continued)

Table 3-11. Trace Header Marker Parameters

Markers Parameter Name	Description
MKR_MWVNA_Xx	Marker x X value (where x is the marker number 1 to 12)
MKR_MWVNA_POINTx	Marker x display point
MKR_MWVNA_REALx	Marker x Real value
MKR_MWVNA_IMAGx	Marker x Imaginary value
MKR_MWVNA_READOUTx	Marker x readout style. Available readout styles are: Graph = 0 Log Mag = 1 Log Mag and Phase = 2 Phase = 3 Real and Imaginary = 4 SWR = 5 Impedance = 6 Admittance = 7 Normalized Impedance = 8 Normalized Admittance = 9 Polar Impedance = 10 Group Delay = 11 Log Mag/2 = 12 Lin Mag = 13 Lin Mag and Phase = 14

MKR_MWVNA_FLAGSx	Marker x flags:
	`MWVNA_MARKER_REF_BIT ..... 0x00000001`
	`MWVNA_MARKER_DELTA_BIT ..... 0x00000002`
	`MWVNA_MARKER_ALL_BIT ..... 0x00000004`
	`MWVNA_MARKER_INIT_BIT ..... 0x00000008`
	`MWVNA_MARKER_ZERO_SPAN_BIT ..... 0x00000010`
	`MWVNA_MARKER_OUT_OF_RANGE_BIT ... 0x00000020`

MKR_TRACEx	Specifies to which trace the marker x is attached
MKR_DELTA_TOx	Specifies to which trace the marker x is delta

Trace Header Transfer (continued)

Table 3-12. Trace Header Limits Parameters

Limits Parameter Name	Description

LIMIT_MWVNA_FLAGS_UPx
LIMIT_MWVNA_FLAGS_LOx
	Upper(UP)/Lower(LO) Limit flags for trace x:
	`LIMIT_LEFT_OF_SCREEN .... 0x00000001`
	`LIMIT_RIGHT_OF_SCREEN ... 0x00000002`
	`LIMIT_IS_ON ..... 0x00000004`
	`LIMIT_IS_RELATIVE ..... 0x00000008`
	`LIMIT_ALARM_IS_ON ..... 0x00000010`
	`LIMIT_LIMIT_UNITIALIZED . 0x00000020`
	`LIMIT_MESSAGE_ON ..... 0x00000040`

LIMIT_MWVNA_POINT_UPx_# LIMIT_MWVNA_POINT_LOx_#	Upper/Lower Limit point value for trace x, where # is the limit point number. Each point value contains the X-axis value, Y-axis value, limit point, and limit flags, separated by a space. **Note:** Limit point and limit flags are not currently used and will always return a value of 0.000000 for limit point and 0 for limit flags.
LIMIT_MWVNA_GRAPH_TYPE_UPx LIMIT_MWVNA_GRAPH_TYPE_LOx	Upper/Lower Limits Graph type. For available graph types, refer to “GRAPH_TYPE”.
LIMIT_MWVNA_TOTAL_POINTS_UPx LIMIT_MWVNA_TOTAL_POINTS_LOx	Upper/Lower Limit total points.

Chapter 4 — Vector Voltmeter Commands

4-1 Introduction

This chapter describes commands for Vector Voltmeter mode. Only the commands that are listed in this chapter and in Chapter 7, “All Mode Commands” can be used in Vector Voltmeter mode. Using commands from other modes may produce unexpected results.

Notational conventions are described in Section 2-10 “Command and Query Notational Conventions” on page 2-12.

4-2 Vector Voltmeter Commands

Table 4-1. VVM Commands Subsystems

Keyword	Parameter Data or Units
`:MMEMory`	“:MMEMory Subsystem” on page 4-3
`:TRACe`	“:TRACe VVM Subsystem” on page 4-8
`[:SENSe]:VVM`	“[:SENSe]:VVM Subsystem” on page 4-12
`[:SENSe]:VVM:CABLe`	“[:SENSe:]:VVM:CABLe Subsystem” on page 4-14
`[:SENSe]:VVM:FREQuenc`	“[:SENSe]:VVM:FREQuency Subsystem” on page 4-15
`[:SENSe]:VVM:REFerence`	“[:SENSe:]:VVM:REFerence Subsystem” on page 4-16
`:FETCh:VVM`	“:FETCh:VVM Subsystem” on page 4-17
`:FETCh:VVM:REFerence`	“FETCh:VVM:REFerence Subsystem” on page 4-20

4-3 VVM Commands

Table 4-2. VVM Commands Subsystems

Keyword	Parameter Data or Units
:MMEMory	Refer to ":MMEMory:STORe Subsystem" on page 4-6
:TRACe	Refer to ":TRACe VVM Subsystem" on page 4-8
[:SENSe]	
:VVM	Refer to "[:SENSe]:VVM Subsystem" on page 4-12
:CABLe	Refer to "[:SENSe:]:VVM:CABLe Subsystem" on page 4-14
:FREQuency	Refer to "[:SENSe]:VVM:FREQuency Subsystem" on page 4-15
:REFerence	Refer to "[:SENSe:]:VVM:REFerence Subsystem" on page 4-16
:FETCh	
:VVM	Refer to ":FETCh:VVM Subsystem" on page 4-17
:REFerence	Refer to "FETCh:VVM:REFerence Subsystem" on page 4-20
The following commands are described in Chapter 3.	
[:SENSe]:CALibration	Refer to "[:SENSe]:CALibration Subsystem" on page 3-66
[:SENSe]:CORRection	Refer to "[:SENSe]:CORRection Subsystem" on page 3-67
[:SENSe]:CORRection :CKIT	Refer to "[:SENSe]:CORRection:CKIT Subsystem" on page 3-70
[:SENSe]:CORRection :CKIT:USER	Refer to "[:SENSe]:CORRection:CKIT:USER Subsystem" on page 3-75
[:SENSe]:CORRection :COLLect	Refer to "[:SENSe]:CORRection:COLLect Subsystem" on page 3-78

4-4 :MMEMory Subsystem

The Mass :MEMory subsystem contains functions that provide access to the instrument setup and data storage.

Table 4-3. :MMEMory Subsystem

Keyword	Parameter Form	Parameter Data or Units	Notes
:MMEMory			
:LOAD		Refer to ":MMEMory:LOAD Subsystem" on page 4-4	
:STORe		Refer to ":MMEMory:STORe Subsystem" on page 4-6	

4-5 :MMEMory:LOAD Subsystem

The Mass MEMory LOAD subsystem contains commands to transfer from the mass memory device to the internal memory.

Recall Setup

:MMEMory:LOAD:STATe

Description: Recalls a previously stored setup from the current storage location. The saved setup that is to be loaded is specified by `<filename>`. `<filename>` must be enclosed in either single quotes (' ') or double quotes (" ") and must include the extension `".stp"`. The `<integer>` parameter is not currently used, but it must be sent. Send a value of `1`.

Note	When recalling setups for modes other than the current mode, Anritsu recommends that you first change modes, then recall the setup. Use ":INSTrument:NSELect" on page 7-3 or ":INSTrument[:SELect]" on page 7-4 to change the instrument state, then use `:MMEMory:LOAD:STATe` to recall the setup.

Syntax: `:MMEMory:LOAD:STATe <integer>, <filename>`

Cmd Parameter: `<integer>, <string> (1, file name)`

Query Response: **NA** (no query)

Related Command: `:MMEMory:STORe:STATe`
`:MMEMory:MSIS INTernal|USB`

Front Panel Access: **Shift-7** (File), Recall

Recall Measurement

:MMEMory:LOAD:TRACe

Description: Recalls a previously stored measurement trace from the current storage location. The saved measurement trace that is to be loaded is specified by `<filename>`. `<filename>` must be enclosed in either single quotes (‘ ’) or double quotes (“ ”) and must contain a file extension of `".mna"`. Note that the trace that is specified by `<filename>` must be available at the current mass storage location. The `<integer>` parameter is not currently in use, but it must be sent. Send a `1`. File Extensions: `".mna"`.

Syntax: `:MMEMory:LOAD:TRACe <integer>, <filename>`

Cmd Parameter: `<integer>, <string> (1, file name)`

Query Response: **NA** (no query)

Example: To recall trace with file name “trace”:

```
:MMEMory:LOAD:TRACe 1,"trace.mna"
```

Related Command: `:MMEMory:STORe:TRACe`
`:MMEMory:MSIS INTernal|USB`

Front Panel Access: **Shift-7** (File), Recall

4-6 :MMEMory:STORe Subsystem

The Mass MEMory STORe subsystem contains commands to transfer from the internal memory to the mass memory device.

Save Setup

:MMEMory:STORe:STATe

Description: Stores the current setup into the file that is specified by `<filename>`. `<filename>` must be enclosed in either single quotes (‘ ’) or double quotes (“ ”) and must not contain a file extension. The `<integer>` is used to distinguish whether the calibration should be saving with the setup. Send a 1 to save setup without a calibration. Send a 2 to save setup with calibration.

Syntax: `:MMEMory:STORe:STATe <integer>, <filename>`

Cmd Parameter: `<integer>, <string> (1|2, filename)`

Query Response: **NA** (no query)

Front Panel Access: **NA**

Save Measurement

:MMEMory:STORe:TRACe

Description: Stores the trace into the file that is specified by `<filename>`. `<filename>` must be enclosed in either single quotes (‘ ’) or double quotes (“ ”) and must not contain a file extension. Note that existing files of the same name will not be overwritten. The `<integer>` parameter is used to distinguish which type of files to save. The following types are available:

<Integer>	:	File type
1	:	Measurement file (default, if number is not 1 to 6)
2	:	S2P Real/Imag
3	:	S2P Lin Mag/Phase
4	:	S2P Log Mag/Phase
5	:	Text
6	:	CSV

Syntax: `:MMEMory:STORe:TRACe <integer>, <filename>`

Cmd Parameter: `<integer>, <string> (1|2, filename)`

Query Response: **NA** (no query)

Example: To save the trace into the file named “trace”.

```
:MMEMory:STORe:TRACe 1,”trace”
```

Related Command: `:MMEMory:LOAD:TRACe`

Front Panel Access: **Shift-7** (File), Save
Shift-7 (File), Save Measurement

4-7 :TRACe VVM Subsystem

This subsystem contains commands pertaining to the Vector Voltmeter mode.

Table 4-4. :TRACe VVM Subsystem Commands

Keyword	Parameter Form	Parameter Data or Units	Notes
`:TRACe`			
`:PREamble?`	<char>	Returns block data	Query Only

Trace Header Transfer

`:TRACe:PREamble?`

Description: Query only. Returns trace header information. The response begins with an ASCII header. The header specifies the number of following bytes. It appears in the format `#AX`, where `A` is the number of digits in `X`, and `X` is the number of bytes that follow the header.

Parameters are returned in comma-delimited ASCII format. Each parameter is returned as `"NAME=VALUE[UNITS]"`. Note that the parameters that are returned depend on the firmware version and that this document does not cover all parameter values that are returned by the command. Refer to Table 4-5, "Trace Header Parameters.

For the example response, the serial number (SN) is 83320012 and is returned as "SN=83320012". Refer to section "Example Response Format:" on page 4-9.

Syntax: `:TRACe:PREamble?`

Query Response: `<char> (returns block data)`

Front Panel Access: **NA**

Example Response Format:

[#800001070SN=83320012,UNIT_NAME=,TYPE=DATA,DATE=1999-11-30-02-00-10-10, APP_NAME=MWVNA,APP_VER=T0.00.1001,VVM_MODE=0.000000,VVM_CW_FREQ= 0.005000,VVM_MEAS_TYPE=0.000000,VVM_RETURN_MEAS_FORMAT=0.000000, VVM_CABLE=1.000000,VVM_PORT_1_SAVE_RETURN_REF=0.000000,VVM_PORT_1_ SAVE_INSERTION_REF=0.000000,VVM_PORT_2_SAVE_RETURN_REF=0.000000,VVM_ PORT_2_SAVE_INSERTION_REF=0.000000,VVM_PORT_1_RETURN_REF_AMP= 0.000000,VVM_PORT_1_RETURN_REF_PHASE=0.000000,VVM_PORT_1_RETURN_REF_ VSWR=1000.000000,VVM_PORT_1_RETURN_REF_REAL=0.000000,VVM_PORT_1_ RETURN_REF_IMAG=0.000000,VVM_PORT_1_INSERTION_REF_AMP=0.000000,VVM_ PORT_1_INSERTION_REF_PHASE=0.000000,VVM_PORT_1_RETURN_REF_RAW_ REAL=1000000.000000,VVM_PORT_1_RETURN_REF_RAW_IMAG=0.000000,VVM_PORT _2_RETURN_REF_AMP=0.000000,VVM_PORT_2_RETURN_REF_PHASE=0.000000,VVM_ PORT_2_RETURN_REF_VSWR=1000.000000,VVM_PORT_2_RETURN_REF_REAL= 0.000000,VVM_PORT_2_RETURN_REF_IMAG=0.000000,VVM_PORT_2_INSERTION_ REF_AMP=0.000000,VVM_PORT_2_INSERTION_REF_PHASE=0.000000,VVM_PORT_2_ RETURN_REF_RAW_REAL=1000000.000000, VVM_PORT_2_RETURN_REF_RAW_IMAG=0.000000, CAL_PORT=1]

Trace Header Parameters

Table 4-5 describes parameters that can be returned by the `:TRACe:PREamble?` command.

Table 4-5. Trace Header Parameters (Sheet 1 of 2)

Parameter Name	Description
SN	Instrument Serial #
UNIT_NAME	Instrument name
DATE	Trace date/time
APP_NAME	Application name
APP_VER	Application firmware (FW) version

VVM_MODE	Mode
	`0 = CW`
	`1 = Table`

VVM_CW_FREQ	CW frequency

VVM_MEAS_TYPE	Measurement Type
	`0 = Return`
	`1 = Insertion`

VVM_RETURN_MEAS_FORMAT	Return Type Measurement Format
	`0 = dB`
	`1 = VSWR`
	`2 = Impedance`

VVM_CABLE	Selected Cable number 1 to 12

Table 4-5. Trace Header Parameters (Sheet 2 of 2)

Parameter Name	Description
VVM_PORT_X_SAVE_RETURN_REF	Saved status for Port x Return reference, where x = 1 or 2
VVM_PORT_X_SAVE_INSERTION_REF	Saved status for Port x Insertion reference, where x = 1 or 2
VVM_PORT_X_RETURN_REF_AMP	Return reference amplitude for Port x, where x = 1 or 2
VVM_PORT_X_RETURN_REF_PHASE	Return reference phase for Port x, where x = 1 or 2
VVM_PORT_X_RETURN_REF_VSWR	Return reference VSWR for Port x, where x = 1 or 2
VVM_PORT_X_RETURN_REF_REAL	Return reference real for Port x, where x = 1 or 2
VVM_PORT_X_RETURN_REF_IMAG	Return reference imaginary for Port x, where x = 1 or 2
VVM_PORT_X_INSERTION_REF_AMP	Insertion reference amplitude for Port x, where x = 1 or 2
VVM_PORT_X_INSERTION_REF_PHASE	Insertion reference phase for Port x, where x = 1 or 2

4-8 [:SENSe]:SWEep Subsystem

This subsystem contains commands pertaining to the Vector Voltmeter mode.

IF Bandwidth

[:SENSe]:SWEep:IFBW

Description: Sets the IF Bandwidth. When the Measurement View is set to VNA, the `<freq value>` in Hz must be one of the following 13 values:

`100000|50000|20000|10000|5000|2000|1000|500|200|100|50|20|10`

When the Measurement view is set to FIELD, the `<freq value>` in Hz must be one of the following 3 values:

`100000|1000|10`

The query form of this command returns the frequency in Hz.

Syntax:
```
[:SENSe]:SWEep:IFBW <freq value>
[:SENSe]:SWEep:IFBW?
```

Cmd Parameter: `<char> <freq value>`

Query Response: `<char> <freq value>`

Default Value: 10000

Default Units: Hz

Range: 100 to 100000 Hz

Example: Sets the IF Bandwidth frequency to 200 Hz:

```
:SENS:SWE:IFBW 200
```

Sets the IF Bandwidth frequency to 100 kHz:

```
:SENS:SWE:IFBW 100000
```

Front Panel Access: **CW** or **Table**, IFBW

4-9 [:SENSe]:VVM Subsystem

This subsystem contains commands pertaining to the Vector Voltmeter mode.

Table 4-6. [:SENSe]:VVM Subsystem Commands

Keyword	Parameter Data or Units
`[:SENSe]`	
`:VVM`	
`:CABLe`	Refer to "[:SENSe:]:VVM:CABLe Subsystem" on page 4-14.
`:FREQuency`	Refer to "[:SENSe]:VVM:FREQuency Subsystem" on page 4-15
`:REFerence`	Refer to "[:SENSe:]:VVM:REFerence Subsystem" on page 4-16

Return Measurement Format

`[:SENSe]:VVM:FORMat`

Description: Sets the VVM Return type measurement format. The query format of the command returns the VVM Return type measurement format.

Syntax:
```
[:SENSe]:VVM:FORMat DB|VSWR|IMPedance
[:SENSe]:VVM:FORMat?
```

Cmd Parameter: `<char> DB|VSWR|IMPedance`

Query Response: `<char> DB|VSWR|IMP`

Default Value: DB

Example: To set the type to VSWR:

```
:SENSe:VVM:FORMat VSWR
```

Front Panel Access: **CW**, Return Meas Format

Measurement Mode

[:SENSe]:VVM:MODE CW|TABLe

Description: Sets the VVM measurement mode. The query format of the command returns the VVM measurement mode.

Syntax:
```
[:SENSe]:VVM:MODE CW|TABLe
[:SENSe]:VVM:MODE?
```

Cmd Parameter: `<char> CW|TABLe`

Query Response: `<char> CW|TABL`

Default Value: CW

Example: To set the mode to Table:

```
:SENSe:VVM:MODE TABLe
```

Front Panel Access: CW: **Hard Key 1**

TABLe: **Hard Key 2**

Measurement Type

[:SENSe]:VVM:TYPE

Description: Sets the VVM measurement type. The query format of the command returns the VVM measurement type.

Syntax:
```
[:SENSe]:VVM:TYPE RETurn|INSertion
[:SENSe]:VVM:TYPE?
```

Parameter: `RETurn|INSertion`

Cmd Parameter: `<char> RETurn|INSertion`

Query Response: `<char> RET|INS`

Default Value: RET

Example: To set the type to Insertion:

```
:SENSe:VVM:TYPE INSertion
```

Front Panel Access: **CW/Table**, Measurement Type

4-10 [:SENSe:]:VVM:CABLe Subsystem

This subsystem contains commands to select and query the VVM cable.

Cable

[:SENSe]:VVM:CABLe:SELect

Description: Selects the VVM cable. The query format of the command returns the current VVM cable number.

Syntax:
```
[:SENSe]:VVM:CABLe:SELect 1|2|3|4|5|6|7|8|9|10|11|12
[:SENSe]:VVM:CABLe:SELect?
```

Cmd Parameter: `<char> 1|2|3|4|5|6|7|8|9|10|11|12`

Query Response: `<char> 1|2|3|4|5|6|7|8|9|10|11|12`

Default Value: 1

Example: To set the Cable to 6:

```
:SENSe:VVM:CABLe:SELect 6
```

Front Panel Access: **Table**, Select Cable

4-11 [:SENSe]:VVM:FREQuency Subsystem

This subsystem contains commands pertaining to the frequency settings of the Vector Voltmeter.

CW Frequency

[:SENSe]:VVM:FREQuency:CW

Description: Sets the VVM CW frequency. The query format of the command returns the CW frequency.

Syntax:
```
[:SENSe]:VVM:FREQuency:CW <freq>
[:SENSe]:VVM:FREQuency:CW?
```

Cmd Parameter: `<NRf> <freq> (hertz)`

Query Response: `<NR3> <freq> (hertz)`

Default Value: 500 kHz

Default Units: Hz

Range: 500 kHz to Unit Maximum Frequency

Front Panel Access: **CW/Table**, CW Frequency

4-12 [:SENSe:]:VVM:REFerence Subsystem

This subsystem contains commands to set and clear the reference VVM data.

Clear Reference

`[:SENSe]:VVM:REFerence:CLEar`

Description: Clears the reference data for the current port and measurement type.

Syntax: `[:SENSe]:VVM:REFerence:CLEar`

Cmd Parameter: **NA**

Query Response: **NA** (no query)

Default Value: No Reference

Example: To clear the Reference:

`:SENSe:VVM:REFerence:CLEar`

Front Panel Access: **CW/Table**, Clear Reference

Set Reference

`[:SENSe]:VVM:REFerence:MEMorize`

Description: Sets the reference data for the current port and measurement type.

Syntax: `[:SENSe]:VVM:REFerence:MEMorize`

Cmd Parameter: **NA**

Query Response: **NA** (no query)

Default Value: No Reference

Example: To set the new Reference:

`:SENSe:VVM:REFerence:MEMorize`

Front Panel Access: **CW/Table**, Save New Reference

4-13 :FETCh:VVM Subsystem

This subsystem contains commands to fetch the VVM reference data and relative data.

Table 4-7. :FETCh:VVM Subsystem Commands

Keyword	Parameter Data or Units
`:FETCh`	
`:VVM`	
`:REFerence`	Refer to "FETCh:VVM:REFerence Subsystem" on page 4-20

Data

:FETCh:VVM:DATA?

Description: Query only. Returns the most recent VVM measurement results. Data is returned as 2 or 4 comma-separated values depending upon the measurement type, measurement format, measurement mode, port, and the reference setting. A "–" is returned for any data that is not valid at that instance.

Table 4-8. VVM Measurement Results (Sheet 1 of 2)

If	Then	Data Values
If the measurement type is Insertion, and if the measurement mode is CW,	then data is returned as 4 comma-separated values	Amplitude Phase Reference Amplitude Reference Phase.
If the measurement type is Insertion, and if the measurement mode is CW with save new reference set,	then data is returned as 4 comma-separated values	Relative Amplitude Relative Phase Reference Amplitude Reference Phase.
If the measurement type is Return, and if the measurement mode is CW, and if format is set to dB,	then data is returned as 4 comma-delimited values	Amplitude Phase Reference Amplitude Reference Phase.
If the measurement type is Return, and if the measurement mode is CW with save new reference set, and if format is set to dB,	then data is returned as 4 comma-delimited values	Relative Amplitude Relative Phase Reference Amplitude Reference Phase.
If the measurement type is Return, and if format is set to VSWR,	then data is returned as 2 comma-delimited values	VSWR Reference VSWR.
If the measurement type is Return, and if the measurement mode is CW with save new reference set, and if format is set to VSWR,	then data is returned as 2 comma-delimited values	Relative VSWR Reference VSWR.
If the measurement type is Return, and if the measurement mode is CW, and if format is set to Impedance,	then data is returned as 4 comma-delimited values	Real Imaginary Reference Real Reference Imaginary.

Table 4-8. VVM Measurement Results (Sheet 2 of 2)

If	Then	Data Values
If the measurement type is Return, and if the measurement mode is CW with save new reference set, and if format is set to Impedance,	then data is returned as 4 comma-delimited values	Relative Real Relative Imaginary Reference Real Reference Imaginary.
If the measurement mode is Table with save new reference set,	then data is returned as 4 comma-separated values	Amplitude Phase Relative Amplitude Relative Phase.

Syntax: `:FETCh:VVM:DATA?`

Cmd Parameter: **NA** (query only)

Query Response: **NA** (comma separated values)

Example: To fetch the VVM data:

`:FETCh:VVM:DATA?`

Front Panel Access: **NA**

4-14 FETCh:VVM:REFerence Subsystem

This subsystem contains commands to fetch VVM reference data.

Reference Data

:FETCh:VVM:REFerence:DATA?

Description: Returns the reference data depending upon the measurement type, the measurement format, and the current port.

Syntax: `:FETCh:VVM:REFerence:DATA?`

Cmd Parameter: **NA** (query only)

Query Response: **NA** (comma separated values)

Example: To fetch the VVM reference data:

```
:FETCh:VVM:REFerence:DATA?
```

Front Panel Access: **NA**

Chapter 5 — Spectrum Analyzer Commands

5-1 Introduction

This chapter describes commands for Spectrum Analyzer mode. Only the commands that are listed in this chapter and in Chapter 7, "All Mode Commands" can be used in Spectrum Analyzer mode. Using commands from other modes may produce unexpected results.

Notational conventions are described in Section 2-10 "Command and Query Notational Conventions" on page 2-12.

Spectrum Analyzer Commands

Table 5-1. SPA Commands Subsystems

Keyword	Parameter Data or Units
`:ABORt`	":ABORt Subsystem" on page 5-2
`:CALCulate`	":CALCulate Subsystem" on page 5-3
`:CONFigure`	":CONFigure Subsystem" on page 5-30
`:DISPlay`	":DISPlay Subsystem" on page 5-33
`:FETCh`	":FETCh Subsystem" on page 5-35
`:FORMat`	":FORMat Subsystem" on page 5-38
`:INITiate`	":INITiate Subsystem" on page 5-40
`:MEASure`	":MEASure Subsystem" on page 5-42
`:MMEMory`	":MMEMory Subsystem" on page 5-46
`:READ`	":READ Subsystem" on page 5-51
`:SENSe`	"[:SENSe] Subsystem" on page 5-71
`:TRACe`	":TRACe Subsystem" on page 5-54
`:TRIGger`	":TRIGger Subsystem" on page 5-68
`:UNIT`	":UNIT Subsystem" on page 5-70
`[:SENSe]`	"[:SENSe] Subsystem" on page 5-71

5-2 :ABORt Subsystem

The abort subsystem includes commands that allow the user to stop current measurement activities on the instrument.

:ABORt

Description: Restarts the current sweep and/or measurement. Resets the trigger system. If `:INITiate:CONTinuous` is OFF (in other words, the instrument is in single sweep mode), then send the command `:INITiate[:IMMediate]` to trigger the next sweep. If `:INITiate:CONTinuous` is ON (in other words, then the instrument is in continuous sweep mode) a new sweep will start immediately.

Syntax: `:ABORt`

Parameter: **NA**

Query Response: **NA** (no query)

Related Command: `:INITiate:CONTinuous`
`:INITiate[:IMMediate]`

Front Panel Access: **Shift-3** (Sweep), Reset Sweep

5-3 :CALCulate Subsystem

The commands in this subsystem process data that has been collected via the SENSe subsystem.

Limit Alarm

:CALCulate:LIMit:ALARm

Description: Enables or disables the currently active limit line alarm. Setting the value to ON or 1 turns on the limit alarm. Setting the value to OFF or 0 turns off the limit alarm. The query version of the command returns a 1 if the currently selected limit line alarm is set to ON and returns 0 if OFF. Use `:CALCulate:LIMit:TYPe` to set the currently active limit line.

Syntax:
```
:CALCulate:LIMit:ALARm OFF|ON|0|1
:CALCulate:LIMit:ALARm?
```

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: OFF

Example: To turn off limit alarm:

```
:CALCulate:LIMit:ALARm OFF
:CALCulate:LIMit:ALARm 0
```

To turn on limit alarm:

```
:CALCulate:LIMit:ALARm ON
:CALCulate:LIMit:ALARm 1
```

Related Command: `:CALCulate:LIMit:TYPe`

Front Panel Access: **Shift-6 (Limit)**, Limit Alarm

Limit Fail

:CALCulate<Tr>:LIMit:FAIL?

Description: Returns the fail status of the given trace `<Tr>` based on the limits. The limit and the limit message must be ON in order to return a valid boolean value. If either or both limits fail, then a `1` is returned.

<Tr> is the trace number in the range 1 to 4. If no trace number is specified, then the <Tr> parameter defaults to trace number 1.

Syntax: :CALCulate<Tr>:LIMit:FAIL?

Cmd Parameter: **NA** (query Only)

Cmd Parameter: <NR1> <integer>

Example: To query the fail status on trace 2:

CALCulate2:LIMit:FAIL?

:CALC2:LIM:FAIL?

Related Command: :CALCulate:LIMit[:STATe]

:CALCulate<Tr>:LIMit:PFMessage

Front Panel Access: **NA**

Lower Limit Fail State

:CALCulate<Tr>:LIMit:LOWer:FAIL?

Description: Returns the fail status of the given trace <Tr>. The lower limit and the limit message must be ON in order to return a valid boolean value. If the lower limit fails, then a 1 is returned. Otherwise, a 0 is returned. <Tr> is the trace number in the range 1 to 4. If no trace number is specified, then the <Tr> parameter defaults to trace number 1.

Syntax: :CALCulate<Tr>:LIMit:LOWer:FAIL?

Cmd Parameter: **NA** (query Only)

Query Parameter: <NR1> <integer>

Example: To query for the fail status on trace 2 for the lower limit:

:CALC2:LIM:LOW:FAIL?

:CALCulate2:LIMit:LOWer:FAIL?

Related Command: :CALCulate<Tr>:LIMit:PFMessage

:CALCulate:LIMit:TYPE

Front Panel Access: **NA**

Limit Pass/Fail

:CALCulate<Tr>:LIMit:PFMessage

Description: Enables or disables the selected trace pass fail message. `<Tr>` is the trace number in the range 1 to 4. If no trace number is specified, then the `<Tr>` parameter defaults to trace number 1. Setting the value to ON or to 1 turns on the selected trace pass fail message. Setting the value to OFF or to 0 turns off the selected trace pass fail message.

The query version of the command returns a 1 if the selected trace pass fail message is set to ON, and the query returns a 0 if the selected trace pass fail message is set to OFF.

Syntax:

```
:CALCulate<Tr>:LIMit:PFMessage OFF|ON|0|1
:CALCulate<Tr>:LIMit:PFMessage?
```

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Parameter: `<boolean> 0|1`

Default Value: OFF or 0 (query returns 0 for OFF)

Example: To turn off pass/fail message:

```
:CALCulate:LIMit:PFMessage OFF
:CALCulate:LIMit:PFMessage 0
:CALC:LIM:PFM 0
```

To turn on pass fail message:

```
:CALCulate:LIMit:PFMessage ON
:CALCulate:LIMit:PFMessage 1
:CALC:LIM:PFM 1
```

Related Command:

```
:CALCulate<Tr>:LIMit:PFMessage
:CALCulate:LIMit:TYPE
```

Front Panel Access: **Shift 6** (Limit), Pass Fail Message

Upper Limit Fail State

:CALCulate<Tr>:LIMit:UPPer:FAIL?

Description: Returns the fail status of the given trace <Tr>. The uppper limit and the limit message must be ON in order to return a valid boolean value. If the upper limit fails, then a 1 is returned. Otherwise, a 0 is returned. <Tr> is the trace number in the range 1 to 4. If no trace number is specified, then the <Tr> parameter defaults to trace number 1. Note that the condition is reset after the end of the sweep. To avoid missing a failing condition, send the command after completing a single sweep.

Syntax: `:CALCulate<Tr>:LIMit:UPPer:FAIL?`

Cmd Parameter: **NA** (query Only)

Query Parameter: `<NR1> <integer>`

Example: To query for the fail status on trace 2 for the upper limit:

```
:CALC2:LIM:UPP:FAIL?
:CALCulate2:LIMit:UPPer:FAIL?
```

Related Command:

```
:CALCulate:LIMit[:STATe]
:CALCulate<Tr>:LIMit:PFMessage
:CALCulate:LIMit:TYPE
```

Front Panel Access: **NA**

Move Limit to Current Center Frequency

:CALCulate:LIMit:CENTer

Description: Moves the center of the current active limit line to the center frequency.

Syntax: `:CALCulate:LIMit:CENTer`

Cmd Parameter:

Query Response: `NA (no query)`

Example: To move the limit to the current center:

```
:CALCulate:LIMit:CENTer
```

Front Panel Access: **Shift-6 (Limit)**, Limit Move, Move Limit to Current Center Freq

Create Limit Envelope

:CALCulate:LIMit:ENVelope:CREate

Title: Create Limit Envelope

Description: Creates a limit envelope. This generates a limit line that formed a mask just above or below the existing signals. Note that this command will turn on the currently selected limit line if it is not already on. Use `:CALCulate:LIMit:TYPe` to set the currently active limit line.

Syntax: `:CALCulate:LIMit:ENVelope:CREate`

Cmd Parameter: `NA`

Query Response: `NA (no query)`

Example: To create a limit envelope:

```
:CALCulate:LIMit:ENVelope:CREate
```

Front Panel Access: Shift-6 (Limit), Limit Envelope, Create Envelope

Limit Envelope Offset

:CALCulate:LIMit:ENVelope:OFFSet

Description: Sets limit envelope offset. This defines how far away from the measured signal the active limit envelope is placed. Use `:CALCulate:LIMit:TYPe` to set the currently active limit line.

Parameter: <amplitude>

Syntax:
```
:CALCulate:LIMit:ENVelope:OFFSet <amplitude>
:CALCulate:LIMit:ENVelope:OFFSet?
```

Cmd Parameter: `<amplitude>`

Query Response: `<amplitude>`

Default Value: 3 dB for upper limit, –3 dB for lower limit

Default Unit: dB

Range: –100 dB to +100 dB

Example: To set the limit envelope offset to 5 dB:

```
:CALCulate:LIMit:ENVelope:OFFSet 5
```

Front Panel Access: Shift-6 (Limit), Limit Envelope, Upper Offset (If Limit is toggled to Upper)
Shift-6 (Limit), Limit Envelope, Lower Offset (If Limit is toggled to Lower)

Number of Limit Envelope Points

:CALCulate:LIMit:ENVelope:POINt

Description: Sets the number of inflection point for the currently active limit envelope. Use :CALCulate:LIMit:TYPe to set the currently active limit line. If the active limit shape is square, the number of inflection points must be even; attempting to set an odd value will result in that value being rounded down to the nearest even number.

Parameter: <number>

Syntax:
```
:CALCulate:LIMit:ENVelope:POINt <number>
:CALCulate:LIMit:ENVelope:POINt?
```

Cmd Parameter: `<number>`

Query Response: `<number>`

Default Value: 21 if limit shape is sloped; 20 if limit shape is square

Range: 2 to 41

Example: To set the number of inflection point to 30:

```
:CALCulate:LIMit:ENVelope:POINt 30
```

Front Panel Access: Shift-6 (Limit), Limit Envelope, Upper Points (If Limit is toggled to Upper)
Shift-6 (Limit), Limit Envelope, Lower Points (If Limit is toggled to Lower)

Limit Envelope Shape

:CALCulate:LIMit:ENVelope:SHAPe

Description: Sets the currently active limit envelope shape.

Syntax:
```
:CALCulate:LIMit:ENVelope:SHAPe SQUare|SLOPe
:CALCulate:LIMit:ENVelope:SHAPe?
```

Cmd Parameter: `<char> SQUare|SLOPe`

Query Response: `<char> SQUare|SLOPe`

Example: To set the limit envelope to a square:

```
:CALCulate:LIMit:ENVelope:SHAPe SQUare
```

Front Panel Access: Shift-6 (Limit), Limit Envelope, Upper Shape (If Limit is toggled to Upper) or Lower Shape (If Limit is toggled to Lower)

Update Limit Envelope Frequency

:CALCulate:LIMit:ENVelope:UPDate:X

Description: Updates limit envelope frequency. Note that this command is valid only if the limit envelope shape is set to a square. Use `:CALCulate:LIMit:TYPe` to set the currently active limit line.

Syntax: `:CALCulate:LIMit:ENVelope:UPDate:X`

Example: To adjust the envelope frequency:

```
:CALCulate:LIMit:ENVelope:UPDate:X
```

Update Limit Envelope Amplitude

:CALCulate:LIMit:ENVelope:UPDate:Y

Description: Updates the amplitude of the current limit without changing the frequencies of the inflection point. Use `:CALCulate:LIMit:TYPe` to set the currently active limit line.

Syntax: `:CALCulate:LIMit:ENVelope:UPDate:Y`

Example: To adjust the limit envelope amplitude:

```
:CALCulate:LIMit:ENVelope:UPDate:Y
```

Front Panel Access: Shift-6 (Limit), Limit Envelope, Update Envelope Amplitude

Limit Fail State

:CALCulate:LIMit:FAIL?

Description: Query whether the currently active limit line (upper or lower) has failed or not. The command returns a 0 on success, 1 on fail, and 2 if the current active limit is OFF or the alarm is OFF. Use `:CALCulate:LIMit:TYPe` to set the currently active limit line.

Syntax: `:CALCulate:LIMit:FAIL?`

Lower Limit Alarm

:CALCulate:LIMit:LOWer:ALARm

Description: Enables/disables the lower limit alarm. It is a combination of the commands `:CALCulate:LIMit:TYPe 1;` `:CALCulate:LIMit:ALARm ON|OFF`. Setting the value to ON or 1 will turn on the lower limit alarm. Setting the value to OFF or 0 will turn off the lower limit alarm. The query version of the command returns a 1 if the lower limit line alarm is ON and returns 0 if OFF. Note that using this command set the lower limit line active for editing.

Syntax: `:CALCulate:LIMit:LOWer:ALARm OFF|ON|0|1`

`:CALCulate:LIMit:LOWer:ALARm?`

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: OFF

Related Command: `:CALCulate:LIMit:ALARm`

Front Panel Access: **Shift-6 (Limit)**, Limit Alarm

Lower Limit Fail State

:CALCulate:LIMit:LOWer:FAIL?

Description: Query whether the lower limit line has failed or not. The command returns a 0 on success, returns a 1 on fail, and returns a 2 if the lower limit line is OFF or if the alarm is OFF.

Syntax: `:CALCulate:LIMit:LOWer:FAIL?`

Number of Lower Limit Points

:CALCulate:LIMit:LOWer:POINt?

Description: Returns the number of points currently in the lower limit line.

Syntax: `:CALCulate:LIMit:LOWer:POINt?`

Default Value: 2

Related Command: `:CALCulate:LIMit:POINt?`

Lower Limit State

:CALCulate:LIMit:LOWer[:STATe]

Description: Turns the lower limit line ON or OFF. It is a combination of the commands `:CALCulate:LIMit:TYPe 1;` `:CALCulate:LIMit:STATe ON|OFF`. The query version of the command returns a 1 if the lower limit line is ON and returns a 0 if OFF.

Syntax: `:CALCulate:LIMit:LOWer[:STATe] OFF|ON|0|1`

`:CALCulate:LIMit:LOWer[:STATe]?`

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: OFF

Example: To turn on lower limit:

```
:CALCulate:LIMit:LOWer ON
:CALCulate:LIMit:LOWer 1
:CALCulate:LIMit:LOWer:STATe ON
```

To turn off lower limit:

```
:CALCulate:LIMit:LOWer OFF
:CALCulate:LIMit:LOWer 0
:CALCulate:LIMit:LOWer:STATe 0
```

Related Command: `:CALCulate:LIMit:ALARm`

Front Panel Access: **Shift-6 (Limit)**, On/Off

Limit Line Type

:CALCulate:LIMit:LTYPe

Description: Sets the currently active limit line type. Absolutes limit lines set the limit inflection points based upon the entered frequencies for each point. Relative limit lines set the limit inflection points relative to the current center frequency.

Syntax: `:CALCulate:LIMit:LTYPe ABSolute|RELative`

`:CALCulate:LIMit:LTYPe?`

Cmd Parameter: `<char> ABSolute|RELative`

Query Response: `<char> ABS|REL`

Default Value: ABSolute

Range: ABSolute|RELative

Example: To set the limit line type to relative:

`:CALCulate:LIMit:LTYPe RELative`

Related Command: `:CALCulate:LIMit[:STATe]`

Front Panel Access: **Shift-6 (Limit)**, Limit Advanced, Limit Line Type

Limit Mirror

:CALCulate:LIMit:MIRRor

Description: Creates a limit mirror. Use `:CALCulate:LIMit:TYPe` to set the currently active limit line.

Syntax: `:CALCulate:LIMit:MIRRor`

Related Command: `:CALCulate:LIMit:TYPe`

Front Panel Access: **Shift-6 (Limit)**, Limit Advanced, Limit Mirror

Add Limit Point

:CALCulate:LIMit:POINt:ADD

Description: Adds a new limit point to the currently active limit line. Use `:CALCulate:LIMit:TYPe` to set the currently active limit line.

Syntax: `:CALCulate:LIMit:POINt:ADD`

Related Command: `:CALCulate:LIMit:TYPe`

Front Panel Access: **Shift-6 (Limit)**, Limit Edit, Add Point

Delete Limit Point

:CALCulate:LIMit:POINt:DELete

Description: Deletes the currently active limit point. The active point becomes the point that is immediately to the left of the point that was deleted. Note that deletion is only valid if there are more than 2 limit points. Use `:CALCulate:LIMit:TYPe` to set the currently active limit line.

Syntax: `:CALCulate:LIMit:POINt:DELete`

Example: To delete the currently active limit point:

```
:CALCulate:LIMit:POINt:DELete
```

Front Panel Access: **Shift-6 (Limit)**, Limit Edit, Delete Point

Next Point Left

:CALCulate:LIMit:POINt:LEFT

Description: Selects the limit point immediately to the left of the active point, making it active for editing or deleting. Use `:CALCulate:LIMit:TYPe` to set the currently active limit line.

Syntax: `:CALCulate:LIMit:POINt:LEFT`

Example: To select the point to the left of the active point:

```
:CALCulate:LIMit:POINt:LEFT
```

Front Panel Access: **Shift-6 (Limit)**, Limit Edit, Next Point Left

Next Point Right

:CALCulate:LIMit:POINt:RIGHt

Description: Selects the limit point immediately to the right of the active point, making it active for editing or deleting. Use `:CALCulate:LIMit:TYPe` to set the currently active limit line.

Syntax: `:CALCulate:LIMit:POINt:RIGHt`

Example: To select the point to the right of the active point:

```
:CALCulate:LIMit:POINt:RIGHt
```

Front Panel Access: **Shift-6 (Limit)**, Limit Edit, Next Point Right

Limit Point X Value

:CALCulate:LIMit:POINt:X

Description: Sets the location of the active limit point on the x-axis at the specified location. `<x-parameter>` is defined in the current x-axis. Note that this will change the Move Limit on the front panel to Point if it is currently set to Limit. The query version of the command returns the location of the active limit point on the x-axis. Use `:CALCulate:LIMit:TYPe` to set the currently active limit line.

Syntax:
```
:CALCulate:LIMit:POINt:X <x-parameter>
:CALCulate:LIMit:POINt:X?
```

Cmd Parameter: `<x-parameter>`

Query Response: `<x-parameter>`

Default Unit: Current x-axis unit, Hz

Example: To set the active point to 5 Hertz:

```
:CALCulate:LIMit:POINt:X 5
:CALCulate:LIMit:POINt:X 5Hz
```

To set the active point to 500 MHz:

```
:CALCulate:LIMIt:POINt:X 500MHz
```

Related Command: `:CALCulate:LIMit:TYPe`

Front Panel Access: **Shift-6 (Limit)**, Limit Edit, Frequency

Limit Point Y Value

:CALCulate:LIMit:POINt:Y

Description: Sets the location of the active limit point on the y-axis at the specified location. `<y-parameter>` is defined in the current y-axis. Note that this will change the Move Limit on the front panel to Point if it is currently set to Limit. The query version of the command returns the location of the active limit point on the y-axis. Use `:CALCulate:LIMit:TYPe` to set the currently active limit line.

Syntax:
```
:CALCulate:LIMit:POINt:Y <y-parameter>
:CALCulate:LIMit:POINt:Y?
```

Cmd Parameter: `<y-parameter>`

Query Response: `<y-parameter>`

Default Unit: Current y-axis unit.

Example: To set the active point to 5 dBm:

```
:CALCulate:LIMit:POINt:Y 5
```

(If y-axis unit is dBm)

`:CALCulate:LIMit:POINt:Y 5dBm`

Front Panel Access: **Shift-6 (Limit)**, Limit Edit, Amplitude

Number of Limit Points

`:CALCulate:LIMit:POINt?`

Description: Returns the number of points currently in the selected limit line. Use `:CALCulate:LIMit:TYPe` to set the currently active limit line.

Syntax: `:CALCulate:LIMit:POINt?`

Related Command: `:CALCulate:LIMit:TYPe`

Set Limit Line Upper or Lower

`:CALCulate:LIMit:TYPe`

Description: Sets the currently active limit line to either upper or lower. Subsequent limit line operations will be performed on the selected limit line.

Syntax: `:CALCulate:LIMit:TYPe 0|1`

`:CALCulate:LIMit:TYPe?`

Cmd Parameter: `<number>`
`0|1 (0 = upper limit line, 1 = lower limit line)`

Query Response: `<number>`

`0|1 (0 = upper limit line, 1 = lower limit line)`

Default Value: 0 (upper)

Range: 0|1

Example: To set the active limit line to upper:

`:CALCulate:LIMit:TYPe 0`

Related Command: None

Front Panel Access: **Shift-6 (Limit)**, Limit

Upper Limit Alarm

:CALCulate:LIMit:UPPer:ALARm

Description: Enables/disables the alarm for the upper limit. It is a combination of the commands `:CALCulate:LIMit:TYPe 0;` `:CALCulate:LIMit:ALARm ON|OFF.` Setting the value to ON or 1 will turn on the upper limit alarm. Setting the value to OFF or 0 will turn off the upper limit alarm. The query version of the command returns a 1 if the upper limit line alarm is ON and returns 0 if OFF.

Syntax: `:CALCulate:LIMit:UPPer:ALARm OFF|ON|0|1`

`:CALCulate:LIMit:UPPer:ALARm?`

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: OFF

Related Command: `:CALCulate:LIMit:ALARm`

Front Panel Access: **Shift-6 (Limit)**, Limit Alarm

Upper Limit Fail State

:CALCulate:LIMit:UPPer:FAIL?

Description: Query whether the upper limit line has failed or not. The command returns a 0 on success, returns a 1 on fail, and returns a 2 if the upper limit line is OFF or the alarm is OFF. Note that the condition is reset after the end of the sweep. To avoid missing a failing condition, send the command after completing a single sweep.

Syntax: `:CALCulate:LIMit:UPPer:FAIL?`

Front Panel Access: **NA**

Number of Upper Limit Points

:CALCulate:LIMit:UPPer:POINt?

Description: Returns the number of points currently in the upper limit line.

Syntax: `:CALCulate:LIMit:UPPer:POINt?`

Default Value: 2

Related Command: `:CALCulate:LIMit:POINt?`

Front Panel Access: **NA**

Upper Limit State

:CALCulate:LIMit:UPPer[:STATe]

Description: Turns the upper limit line ON or OFF. It is a combination of the commands `:CALCulate:LIMit:TYPe 0;` `:CALCulate:LIMit:STATe ON|OFF`. The query version of the command returns a 1 if the upper limit line is ON and returns a 0 if OFF.

Syntax: `:CALCulate:LIMit:UPPer[:STATe] OFF|ON|0|1`

`:CALCulate:LIMit:UPPer[:STATe]?`

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<boolean> 0|1`

Default Value: OFF

Example: To turn on upper limit:

```
:CALCulate:LIMit:UPPer ON
:CALCulate:LIMit:UPPer 1
:CALCulate:LIMit:UPPer:STATe ON
```

To turn off upper limit:

```
:CALCulate:LIMit:UPPer OFF
:CALCulate:LIMit:UPPer 0
:CALCulate:LIMit:UPPer:STATe 0
```

Related Command: `:CALCulate:LIMit[:STATe]`

Front Panel Access: **Shift-6 (Limit)**, On/Off

Move Limit

:CALCulate:LIMit:VALue

Description: Sets the currently active limit line value. This command moves an entire single or multi-segment limit line up or down by the given `<value>`. If the front panel Move Limit button (Shift-6, Limit Move, Move Limit) is pressed, the limit line will move to the given <value>. This command is equivalent to the command `:CALCulate:LIMit:Y`. Use `:CALCulate:LIMit:TYPe` to set the currently active limit line.

Note that this will change the Move Limit on the front panel to Limit if it is currently set to Point.

Syntax: `:CALCulate:LIMit:VALue <value>`

Cmd Parameter: `<value>`

Query Response: `<value>`

Default Unit: Current y-axis unit.

Related Command: `:CALCulate:LIMit:Y`
`:CALCulate:LIMit:TYPe`

Front Panel Access: **Shift-6** (Limit), Limit Edit, Amplitude

Add Vertical

:CALCulate:LIMit:VERTical:ADD

Description: Adds vertical. This will add two inflection points that share the same frequency and are centered midpoint between adjacent points. Use `:CALCulate:LIMit:TYPe` to set the currently active limit line.

Syntax: `:CALCulate:LIMit:VERTical:ADD`

Related Command: `:CALCulate:LIMit:TYPe`

Front Panel Access: **Shift-6** (Limit), Limit Edit, Add Vertical

Limit X Value

:CALCulate:LIMit:X

Description: Sets the location of the active limit line on the x-axis at the specified location. This moves the entire limit and moves the active limit point to the given value. `<x-parameter>` is defined in the current x-axis. Note that this will change the Move Limit on the front panel to Limit if it is currently set to Point. Use `:CALCulate:LIMit:TYPe` to set the currently active limit line.

Syntax: `:CALCulate:LIMit:X <x-parameter>`

Cmd Parameter: `<x-parameter>`

Query Response: `<x-parameter> (limit must exist or an error or -400 is returned)`

Default Unit: Hz or for zero span in seconds

Example: To move the limit and set active point to 5 Hz:

```
:CALCulate:LIMit:X 5
:CALCulate:LIMit:X 5Hz
```

To move the limit and set active point to 500 MHz:

```
:CALCulate:LIMit:X 500MHz
```

To move the limit and set active point to 2.5 seconds (In zero span):

```
:CALCulate:LIMit:X 2.5
:CALCulate:LIMit:X 2.5s
```

To move the limit and set active point to 25 microseconds (In zero span):

```
:CALCulate:LIMit:X 25us
```

Front Panel Access: **Shift-6** (Limit), Limit Edit, Frequency

Limit Line Y Value

:CALCulate:LIMit:Y

Description: Sets the location of the active limit line on the y-axis at the specified location. This moves the entire limit and moves the current active limit point by the given value. `<y-parameter>` is defined in the current y-axis. Note that this will change the Move Limit on the front panel to Limit if it is currently set to Point. Use `:CALCulate:LIMit:TYPe` to set the currently active limit line.

Syntax: `:CALCulate:LIMit:Y <y-parameter>`

Cmd Parameter: `<y-parameter>`

Query Response: `<y-parameter> (limit must exist or an error or -400 is returned)`

Default Unit: Current y-axis unit.

Example: To move limit and set the active point to 5 dbm:

```
:CALCulate:LIMit:Y 5
```

(If y-axis unit is dBm)

```
:CALCulate:LIMit:Y 5dBm
```

Front Panel Access: **Shift-6** (Limit), Limit Edit, Amplitude

Set Default Limit

:CALCulate:LIMit[:SET]:DEFault

Description: Deletes all limit points for the currently active limit line and sets the default limit line value. Note that this command will turn on the currently selected limit line if it is not already on. The current selected limit line can be modified by using the command `:CALCulate:LIMit:TYPe.`

Syntax: `:CALCulate:LIMit[:SET]:DEFault`

Example: To set the default limit line:

```
:CALCulate:LIMit[:SET]:DEFault
:CALCulate:LIMit:DEFault
```

Front Panel Access: **Shift-6** (Limit), Set Default Limit

Limit State

:CALCulate:LIMit[:STATe]

Description: Turns the currently selected limit line (upper or lower) ON or OFF. If the value is set to ON or 1, then the currently selected limit line is ON. If the value is set to OFF or 0, then the currently selected limit line is OFF. The query version of the command returns a 1 if the currently selected limit line is ON and returns a 0 if OFF. Use `:CALCulate:LIMit:TYPe` to set the currently active limit line.

Syntax: `:CALCulate:LIMit[:STATe] OFF|ON|0|1`

`:CALCulate:LIMit[:STATe]?`

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: OFF

Example: To turn on the currently selected limit line:

```
:CALCulate:LIMit ON
:CALCulate:LIMit:STATe ON
:CALCulate:LIMit:STATe 1
```

To turn off the currently selected limit line:

```
:CALCulate:LIMit OFF
:CALCulate:LIMit:STATe 0
:CALCulate:LIMit 0
```

Front Panel Access: **Shift-6** (Limit), On/Off

Turn All Markers Off

:CALCulate:MARKer:AOFF

Description: Turns off all markers.

Syntax: `:CALCulate:MARKer:AOFF`

Front Panel Access: Marker, More, All Markers Off

Peak Threshold

:CALCulate:MARKer:PEAK:THReshold

Description: Sets the peak/valley threshold as a percentage of the display. `:CALCulate:MARKer:MAXimum:LEFT` and `:CALCulate:MARKer:MAXimum:RIGHt` use this value to determine whether a particular display point qualifies as a peak.

Syntax: `:CALCulate:MARKer:PEAK:THReshold <percentage>`

`:CALCulate:MARKer:PEAK:THReshold?`

Cmd Parameter: `<percentage>`

Query Response: `<percentage>`

Default Value: 10

Default Unit: %

Range: 0% to 100%

Front Panel Access: Marker, More Peak Options, Peak Threshold

Marker Data

:CALCulate:MARKer:TABLe:DATA?

Description: Returns the marker table contents.

Syntax: `:CALCulate:MARKer:TABLe:DATA?`

Cmd Parameter: **NA** (query only)

Query Response: **NA**

Front Panel Access: **Marker**, More, Marker Table On

Marker Table State

:CALCulate:MARKer:TABLe[:STATe]

Description: Turns the Marker Table on or off. Setting the value to ON turns on the marker table. Setting the value to OFF turns off the marker table.

Syntax:
```
:CALCulate:MARKer:TABLe[:STATe] OFF|ON
:CALCulate:MARKer:TABLe[:STATe]?
```

Cmd Parameter: `<boolean> OFF|ON`

Query Response: `<boolean> OFF|ON`

Default Value: OFF

Example: To turn on marker table:

```
:CALCulate:MARKer:TABLe ON
```

Front Panel Access: **Marker**, More, Marker Table

Marker Noise

:CALCulate:MARKer{1|2|3|4|5|6}:DELTa:NOISe[:STATe]

Description: Turns the delta marker noise on or off. Note that if counter marker is set to on when setting marker noise to on, then counter marker is set to off. This command is not valid in zero span. The query version of this command returns a 1 if the specified delta marker is noise marker and returns a 0 if not.

Syntax:
```
:CALCulate:MARKer{1|2|3|4|5|6}:DELTa:NOISe[:STATe]
OFF|ON|0|1
:CALCulate:MARKer{1|2|3|4|5|6}:DELTa:NOISe[:STATe]?
```

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: OFF

Example: To turn on marker noise for delta marker #1:

```
:CALCulate:MARKer1:DELTa:NOISe ON
:CALCulate:MARKer1:DELTa:NOISe 1
:CALCulate:MARKer:DELTa:NOISe 1
:CALCulate:MARKer:DELTa:NOISe:STATe ON
```

To turn on marker noise for delta marker #2:

```
:CALCulate:MARKer2:DELTa:NOISe ON
:CALCulate:MARKer2:DELTa:NOISe 1
:CALCulate:MARKer2:DELTa:NOISe:STATe ON
```

To turn off marker noise #5:

```
:CALCulate:MARKer5:DELTa:NOISe OFF
:CALCulate:MARKer5:DELTa:NOISe 0
:CALCulate:MARKer5:DELTa:NOISe:STATe OFF
```

Front Panel Access: **Marker**, More, Marker Noise

Delta Marker X Value

:CALCulate:MARKer{1|2|3|4|5|6}:DELTa:X

Description: Sets the location of the delta marker on the x-axis at the specified location `<x-parameter>` + the reference marker x-axis. `<x-parameter>` is defined in the current x-axis units. The query version of the command returns the location of the delta marker on the x-axis.

Syntax:

```
:CALCulate:MARKer{1|2|3|4|5|6}:DELTa:X <x-parameter>
:CALCulate:MARKer{1|2|3|4|5|6}:DELTa:X?
```

Cmd Parameter: `<x-parameter>`

Query Response: `<x-parameter>`

Default Unit: Hz or seconds if in zero span

Example: If both the reference and delta marker #1 is currently at 1 GHz on the x-axis, send the command below to set the delta marker #1 to 2 GHz on the x-axis:

```
:CALCulate:MARKer1:DELTa:X 1GHz
```

(In zero span) If both the reference and delta marker #1 is currently at 25 microseconds on the x-axis, then send the following command to set the delta marker to 50 µs on the x-axis:

```
:CALCulate:MARKer1:DELTa:X 25µs
```

Related Command: `:CALCulate:MARKer[1]|2|3|4|5|6:X`

Front Panel Access: **Marker**, Delta

Delta Marker Read Y Value

:CALCulate:MARKer{1|2|3|4|5|6}:DELTa:Y?

Description: Reads the current absolute Y value for the specified delta marker. The units are the units of the y-axis.

Syntax:

```
:CALCulate:MARKer{1|2|3|4|5|6}:DELTa:Y?
```

Default Unit: Current y-axis unit

Front Panel Access: **NA**

Delta Marker to Span

:CALCulate:MARKer{1|2|3|4|5|6}:DELTa[:SET]:SPAN

Description: Sets the total span width to the value of the specified delta marker. Note that this command is valid only if delta marker is on.

Syntax: `:CALCulate:MARKer{1|2|3|4|5|6}:DELTa[:SET]:SPAN`

Example: To set the span to the value of delta marker #4:

```
:CALCulate:MARKer4:DELTa:SPAN
```

Front Panel Access: **Marker**, More Peak Options, Delta Marker to Span

Delta Marker State

:CALCulate:MARKer{1|2|3|4|5|6}:DELTa[:STATe]

Description: Sets the specified delta marker on or off.

Syntax: `:CALCulate:MARKer{1|2|3|4|5|6}:DELTa[:STATe] OFF|ON|0|1`

`:CALCulate:MARKer{1|2|3|4|5|6}:DELTa[:STATe]?`

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: OFF

Example: To turn on delta marker #3:

```
:CAL1Culate:MARKer3:DELTa ON
:CALCulate:MARKer3:DELTa 1
:CALCulate:MARKer3:DELTa:STATe ON
:CALCulate:MARKer3:DELTa:STATe 1
```

To turn off delta marker #6

```
:CALCulate:MARKer6:DELTa OFF
:CALCulate:MARKer6:DELTa:STATe OFF
:CALCulate:MARKer6:DELTa:STATe 0
```

Front Panel Access: **Marker**, Delta

Marker Counter

:CALCulate:MARKer{1|2|3|4|5|6}:FCOunt[:STATe]

Description: Turns the marker frequency counter on or off. The marker counter is turned off when the selected marker is turned off. If delta marker is on when setting marker counter to on, then delta marker is turned off. If noise marker is set to on when setting marker counter to on, then noise marker is set to off. This command is not valid in zero span.

Syntax:
```
:CALCulate:MARKer{1|2|3|4|5|6}:FCOunt[:STATe]
OFF|ON|0|1
```
```
:CALCulate:MARKer{1|2|3|4|5|6}:FCOunt[:STATe]?
```

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: OFF

Example: To turn on frequency counter for reference marker #2:

```
:CALCulate:MARKer2:FCOunt ON
:CALCulate:MARKer2:FCOunt 1
```

Front Panel Access: **Marker**, More, Counter Marker

Marker Fixed State

:CALCulate:MARKer{1|2|3|4|5|6}:FIXed[:STATe]

Description: Sets the specified reference marker fixed state on or off. If Fixed is set to on, then the selected reference markers stay at the currently-set amplitude when the marker is set to Fixed.

Syntax:
```
:CALCulate:MARKer{1|2|3|4|5|6}:FIXed[:STATe] OFF|ON|0|1
```
```
:CALCulate:MARKer{1|2|3|4|5|6}:FIXed[:STATe]?
```

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: OFF

Example: To set reference marker #1 to fixed:

```
:CALCulate:MARKer:FIXed ON
:CALCulate:MARKer:FIXed 1
```

Front Panel Access: **Marker**, More, Marker Style (Fixed)

Marker (Maximum) Peak Search

:CALCulate:MARKer{1|2|3|4|5|6}:MAXimum

Description: Puts the specified marker at the maximum amplitude in the trace.

Syntax: `:CALCulate:MARKer{1|2|3|4|5|6}:MAXimum`

Front Panel Access: **Marker**, Marker [1/2/3/4/5/6], Peak Search
Marker, Marker [1/2/3/4/5/6], More Peak Options, Peak Search

Marker (Maximum) Peak Search Left

:CALCulate:MARKer{1|2|3|4|5|6}:MAXimum:LEFT

Description: Puts the specified marker on the next highest peak to the left of the current peak. The next highest peak must be above the peak threshold. If no point meets that criterion, the marker is set to the first point on the trace.

Syntax: `:CALCulate:MARKer{1|2|3|4|5|6}:MAXimum:LEFT`

Related Command: `:CALCulate:MARKer:PEAK:THReshold`

Front Panel Access: **Marker**, More Peak Options, Next Peak Left

Marker (Maximum) Peak Search Next

:CALCulate:MARKer{1|2|3|4|5|6}:MAXimum:NEXT

Description: Moves the marker to the highest peak anywhere in the trace which is lower than the current marker. If the given marker is not on, the command turns it on and sets it to the second highest peak in the trace. The command uses the existing peak threshold values to determine what is a peak and what is not.

Syntax: `:CALCulate:MARKer{1|2|3|4|5|6}:MAXimum:NEXT`

Related Command: `:CALCulate:MARKer:PEAK:THReshold`

Front Panel Access: None

Marker (Maximum) Peak Search Right

:CALCulate:MARKer{1|2|3|4|5|6}:MAXimum:RIGHt

Description: Puts the specified marker on the next highest peak to the right of the current peak. The next highest peak must be above the peak threshold. If no point meets that criterion, the marker is set to the last point on the trace.

Syntax: `:CALCulate:MARKer{1|2|3|4|5|6}:MAXimum:RIGHt`

Related Command: `:CALCulate:MARKer:PEAK:THReshold`

Front Panel Access: **Marker**, More Peak Options, Next Peak Right

Marker Noise

:CALCulate:MARKer{1|2|3|4|5|6}:NOISe[:STATe]

Description: Turns the marker noise on or off for the specified reference marker. Note that if counter marker is set to on when setting marker noise to on, then counter marker is set to off. This command is not valid in zero span.

Syntax: `:CALCulate:MARKer{1|2|3|4|5|6}:NOISe[:STATe] OFF|ON|0|1`

`:CALCulate:MARKer{1|2|3|4|5|6}:NOISe[:STATe]?`

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: OFF

Example: To set reference marker #3 as noise marker:

```
:CALCulate:MARKer3:NOISe ON
:CALCulate:MARKer3:NOISe 1
```

Front Panel Access: **Marker**, More, Marker Noise

Marker X Value

:CALCulate:MARKer{1|2|3|4|5|6}:X

Description: Sets the location of the marker on the x-axis at the specified location. `<x-parameter>` is defined in the current x-axis units. The query version of the command returns the location of the marker on the x-axis. Note that the marker is snapped to the data point closest to the specified value. If the specified marker is not on, then it is set to on.

Syntax:
```
:CALCulate:MARKer{1|2|3|4|5|6}:X <x-parameter>
:CALCulate:MARKer{1|2|3|4|5|6}:X?
```

Cmd Parameter: `<x-parameter>`

Query Response: `<x-parameter>`

Default Unit: Hz or seconds if in zero span

Example: To set reference marker #2 to 5 hertz on the x-axis:

```
:CALCulate:MARKer2:X 5
:CALCulate:MARKer2:X 5Hz
```

To set reference marker #1 to 1.5 GHz on the x-axis:

```
:CALCulate:MARKer:X 1.5GHz
:CALCulate:MARKer1:X 1.5GHz
```

(In zero span) To set reference marker #3 to 1.5 seconds on the x-axis:

```
:CALCulate:MARKer3:X 1.5
:CALCulate:MARKer3:X 1.5s
```

(In zero span) To set reference marker #1 to 25 microseconds:

```
:CALCulate:MARKer:X 25µs
:CALCulate:MARker1:X 25µs
```

Front Panel Access: **Marker**, Marker [1/2/3/4/5/6]

Marker Read Y Value

:CALCulate:MARKer{1|2|3|4|5|6}:Y?

Description: Reads the current Y value for the specified marker. The units are the units of the y-axis.

Syntax:
```
:CALCulate:MARKer{1|2|3|4|5|6}:Y?
```

Cmd Parameter: **NA** (query only)

Query Response: **NA**

Default Unit: Current y-axis unit

Front Panel Access: **NA**

Marker Frequency to Center

:CALCulate:MARKer{1|2|3|4|5|6}[:SET]:CENTer

Description: Sets the center frequency equal to the frequency of the specified marker. Note that this will result in a change to the start and stop frequencies and may also result in a change to the span. Note that this command is not valid in zero span.

Syntax: `:CALCulate:MARKer{1|2|3|4|5|6}[:SET]:CENTer`

Front Panel Access: **Marker**, Marker Freq to Center

Marker to Reference Level

:CALCulate:MARKer{1|2|3|4|5|6}[:SET]:RLEVel

Description: Sets the reference level equal to the measured amplitude of the specified marker. Note that this may result in a change to the input attenuation.

Syntax: `:CALCulate:MARKer{1|2|3|4|5|6}[:SET]:RLEVel`

Front Panel Access: **Marker**, Marker to Ref Lvl

Marker State

:CALCulate:MARKer{1|2|3|4|5|6}[:STATe]

Description: Sets the specified marker on/off.

Syntax: `:CALCulate:MARKer{1|2|3|4|5|6}[:STATe] OFF|ON|0|1`

`:CALCulate:MARKer{1|2|3|4|5|6}[:STATe]?`

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: OFF

Example: To turn off reference marker #1:

`:CALCulate:MARKer1:STATe OFF`

Front Panel Access: **Marker**, On/Off

5-4 :CONFigure Subsystem

This set of commands prepares the instrument for the selected measurement. It disables any currently-enabled measurements and activates the specified measurement. It sets the instrument to single sweep mode, waiting for an `:INITiate` command. It will not initiate the taking of a measurement.

Current instrument settings may be changed to default values. These changes are identified with their respective measurement commands.

Configure Adjacent Channel Power Ratio

:CONFigure:ACPower

Description: Configures the default adjacent channel power ratio measurement. Disables any other active one-button measurements, including channel power, occupied bandwidth, AM/FM demodulation and C/I. Sets the main channel bandwidth equal to the span. Sets the adjacent channel bandwidth and channel spacing equal to the main channel bandwidth. Sets the detection method to RMS. Sets the instrument to single sweep mode (`:INITiate:CONTinuous OFF`). Measurement settings can be modified by using the `[:SENSe]:ACPower` commands before initiating a sweep.

Syntax: `:CONFigure:ACPower`

Related Command:
`[:SENSe]:ACPower:STATe`
`[:SENSe]:ACPower:BANDwidth|BWIDth:MAIN`
`[:SENSe]:ACPower:BANDwidth|BWIDth:ADJacent`
`[:SENSe]:ACPower:BANDwidth|BWIDth:SPACing`

Front Panel Access: **NA**

Configure Channel Power

:CONFigure:CHPower

Description: Configures the default channel power measurement. Disables any other active one-button measurements, including ACPR, occupied bandwidth, AM/FM demodulation, and C/I. Sets the integration bandwidth equal to the span. Sets the detection method to RMS. Sets the instrument to single sweep mode (`:INITiate:CONTinuous OFF`). Measurement settings can be modified by using the `[:SENSe]:CHPower` commands before initiating a sweep. Note that this measurement is not valid in zero span.

Syntax: `:CONFigure:CHPower`

Related Command:
`[:SENSe]:CHPower:STATe`
`:SENSe:CHPower:BANDwidth|BWIDth:INTegration`

Front Panel Access: **NA**

Configure Field Strength

:CONFigure:FSTRength

Description: Configures the default field strength measurement. Disables any other active one-button measurements, including channel power, adjacent channel power, occupied bandwidth, AM/FM demodulation, and C/I. Sets the antenna to the first antenna in the instrument's antenna list. Sets the instrument to single sweep mode (`:INITiate:CONTinuous OFF`). Measurement settings can be modified by using the `[:SENSe]:FSTRength` commands before initiating a sweep. Note that this measurement is not valid in zero span.

Syntax: `:CONFigure:FSTRength`

Related Command: `[:SENSe]:FSTRength:ANTenna`

Front Panel Access: **NA**

Configure Occupied Bandwidth

:CONFigure:OBWidth

Description: Configures the default occupied bandwidth measurement. Disables any other active one-button measurements, including channel power, ACPR, AM/FM demodulation, and C/I. Sets the method to %. Sets the % of power to 99%. Sets the instrument to single sweep mode (`:INITiate:CONTinuous OFF`). Measurement settings can be modified by using the `[:SENSe]:OBWidth` commands before initiating a sweep. Note that this measurement is not valid in zero span.

Syntax: `:CONFigure:OBWidth`

Related Command:
`[:SENSe]:OBWidth:STATe`
`[:SENSe]:OBWidth:METHod`
`[:SENSe]:OBWidth:PERCent`
`[:SENSe]:OBWidth:XDB`

Front Panel Access: **NA**

Measurement Mode

:CONFigure?

Description: Returns the Measurement mode: “CHP” for channel power, “FLDS” for field strength, “OBW” for occupied bandwidth, “ACP” for ACPR, “AMFM” for AM/FM Demod, “CI” for C/I

Syntax: `:CONFigure?`

Cmd Parameter: **NA** (query only)

Query Response: **NA**

Front Panel Access: **NA**

5-5 :DISPlay Subsystem

This subsystem provides commands that modify the display of data for the user. They do not modify the way in which data are returned to the controller.

Display Grid

:DISPlay:GRID

Description: Turns the sweep window grid lines On or Off.

Syntax: `:DISPlay:GRID 0|1|ON|OFF`

Cmd Parameter: `<boolean> 0|1|ON|OFF`

Query Response: **NA** (no query)

Front Panel Access: **NA**

Auto Reference Level

:DISPlay:WINDow:TRACe:Y:ADJust

Title: Adjust Range

Description: Automatically adjusts reference level if input signal strength is too high (ADC error) or too low.

Example: `:DISPlay:WINDow:TRACe:Y:ADJust`

Front Panel Access: Amplitude, Auto Ref Level

Scale

:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision

Description: Sets the scale (dB/division) for the y-axis.

Syntax: `:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision <rel ampl>`

`:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision?`

Cmd Parameter: `<rel ampl>`

Query Response: `<rel ampl>`

Default Value: 10 dB/div

Default Unit: dB

Range: 1 dB to 15 dB

Front Panel Access: **Amplitude**, Scale

Reference Level

:DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel

Description: Sets the reference level amplitude value for the y-axis. Note that this may cause a change in attenuation if the automatic input attenuation coupling is enabled.

Syntax: `:DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel <amplitude>`

`:DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel?`

Cmd Parameter: `<amplitude>`

Query Response: `<amplitude>`

Default Value: 10 dBm

Default Unit: Current active amplitude unit

Range: With reference level offset = 0 dB: 30 dBm to –130 dBm

Example: To set the reference level to 15 dBm (If y-axis is dBm)

```
:DISPlay:WINDow:TRACe:Y:SCALe:RLEVel 15
:DISPlay:WINDow:TRACe:Y:SCALe:RLEVel 15dBm
```

Related Command: `:DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel:OFFset`

Front Panel Access: Amplitude, Reference Level

Reference Level Offset

:DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel:OFFSet

Description: Sets the reference level offset value for the y-axis.

Syntax: `:DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel:OFFSet <rel ampl>`

`:DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel:OFFSet?`

Cmd Parameter: `<rel ampl>`

Query Response: `<rel ampl>`

Default Value: 0 dB

Default Unit: dB

Range: –100 dB to +100 dB

Front Panel Access: **Amplitude**, RL Offset

5-6 :FETCh Subsystem

This set of commands returns the most recent measurement data of the active measurement. They will not switch to another measurement.

To make a new measurement, use the `INITiate` command. To get new measurement data, use the `READ` or `MEASure` query commands.

Fetch Adjacent Channel Power Ratio

`:FETCh:ACPower?`

Description: Returns the most recent adjacent channel power ratio measurement results. If the instrument is sweeping, it will not return until the sweep is complete. If the instrument is not sweeping and the current data is not valid it will return error –230. This could occur if there was a `*RST` immediately before the `:FETCh?` or if a measurement parameter was changed without an `:INITiate`.

Data is returned as 5 comma-separated values: main channel power, lower adjacent channel power, upper adjacent channel power, lower alternate channel power, upper alternate channel power.

Syntax: `:FETCh:ACPower?`

Default Unit: Current amplitude units

Front Panel Access: **NA**

Fetch Channel Power

`:FETCh:CHPower:CHPower?`

Description: Returns the most recent channel power measurement result. It returns only the channel power, not the channel power density. Use `:FETCh:CHPower?` to get both channel power and channel power density.

Syntax: `:FETCh:CHPower:CHPower?`

Cmd Parameter: **NA** (query only)

Query Response: **NA**

Default Unit: Current amplitude units

Related Command: `:FETCh:CHPower?`
`:FETCh:CHPower:DENSity?`

Front Panel Access: **NA**

Fetch Channel Power Density

:FETCh:CHPower:DENSity?

Description: Returns the most recent channel power density measurement result. It returns only the channel power density, not the channel power. Use `:FETCh:CHPower?` to get both channel power and channel power density. If the instrument is sweeping, it will not return until the sweep is complete. If the instrument is not sweeping, and if the current data is not valid, then it will return error –230. This could occur if there were a `*RST` immediately before the `:FETCh?`, or if a measurement parameter were changed without an `:INITiate`.

Syntax: `:FETCh:CHPower:DENSity?`

Cmd Parameter: **NA** (query only)

Query Response: **NA**

Default Unit: Current amplitude units

Front Panel Access: **NA**

Fetch Channel Power/Density

:FETCh:CHPower?

Description: This command returns the most recent channel power measurement results: channel power and channel power density. If the instrument is sweeping, then it will not return until the sweep is complete. If the instrument is not sweeping and if the current data is not valid, then it will return error –230. This could occur if there were a `*RST` immediately before the `:FETCh?`, or if a measurement parameter were changed without an `:INITiate`.

Data is returned as 2 comma-separated values: channel power, channel power density.

Syntax: `:FETCh:CHPower?`

Cmd Parameter: **NA** (query only)

Query Response: **NA**

Default Unit: Current amplitude units

Related Command: `:FETCh:CHPower:CHPower?`
`:FETCh:CHPower:DENSity?`

Front Panel Access: **NA**

Fetch Occupied Bandwidth Frequency

:FETCh:OBWidth:FREQuency?

Description: Returns the most recent occupied bandwidth lower frequency and upper frequency.

Data is returned as 2 comma-separated values: lower frequency and upper frequency in Hz.

Syntax: `:FETCh:OBWidth:FREQuency?`

Cmd Parameter: **NA** (query only)

Query Response: **NA**

Default Unit: Hz

Related Command: `:FETCh:CHPower?`
`:FETCh:CHPower:CHPower?`

Front Panel Access: **NA**

Fetch Occupied Bandwidth

:FETCh:OBWidth?

Description: Returns the most recent occupied bandwidth measurement results: occupied bandwidth, percent of power, and dB down. One of either percent of power or dB down is measured, and the other is set. That is determined by the value that is set using `[:SENSe]:OBWidth:METHod`. If the instrument is sweeping, then it will not return until the sweep is complete. If the instrument is not sweeping, and if the current data is not valid, then it will return error –230. This could occur if there were a `*RST` immediately before the `:FETCh?`, or if a measurement parameter were changed without an `:INITiate`.

Data is returned as 3 comma-separated values: occupied bandwidth, percent of power, dB down.

Syntax: `:FETCh:OBWidth?`

Cmd Parameter: **NA** (query only)

Query Response: **NA**

Default Unit: OBW in Hz, Percent of Power in %, dB Down in dB

Front Panel Access: **NA**

5-7 :FORMat Subsystem

This subsystem contains commands that determine the formatting of numeric data when it is transferred.

The format setting affects data in specific commands only. If a command is affected, it is noted in the command description.

Numeric Data Format

:FORMat[:READings][:DATA]

Description: This command specifies the format in which data is returned in certain commands. The optional `<length>` parameter is needed for REAL format only. It defines the length of the floating point number in bits. Valid values are 32 and 64. If no length is specified, then the default length of REAL data is set to 64 bits.

`ASCii` format returns the data in comma-separated ASCII format. The units are the current instrument units. This format requires many more bytes, and it is therefore the slowest format. INTeger,32 values are signed 32-bit integers in little-endian byte order. This format returns the data in 4-byte blocks. The units are always mdBm. For example, if the measured result were –12.345 dBm, then that value would be sent as –12345. REAL,32 values are 32-bit floating point numbers conforming to the IEEE 754 standard in little-endian byte order. This format returns the data in 4-byte binary format. The units are the current instrument units. REAL,64 values are 64-bit floating point numbers conforming to the IEEE 754 standard in little-endian byte order. This format returns the data in 8-byte binary format. The units are the current instrument units. For a more precise reading, REAL,64 should be used instead of REAL,32 when the current instrument unit is set to Volt or Watt.

Both INTeger and REAL formats return a definite block length. Each transfer begins with an ASCII header such as #42204 for INTeger,32 and REAL,32 and #44408 for REAL,64. The first digit represents the number of following digits in the header (in this example, 4). The remainder of the header indicates the number of bytes that follow the header (in this example, 2204 for INT,32 and REAL,32 and 4408 for REAL,64). You then divide the number of following bytes by the number of bytes in the data format that you have chosen (4 for both INTeger,32 and REAL,32, and 8 for REAL,64) to get the number of data points (in this example, 551).

Syntax: `:FORMat[:READings][:DATA] ASCii|INTeger,32|REAL,[<length>]`

`:FORMat[:READings][:DATA]?`

Cmd Parameter: `<char> ASCii|INTeger,32|REAL,[<length>]`

Query Response: `<char> ASCii|INTeger,32|REAL,[<length>]`

Default Value: ASCii

Related Command: `:TRACe[:DATA]`

Front Panel Access: **NA**

5-8 :INITiate Subsystem

This subsystem controls the triggering of measurements.

Continuous/Single Sweep

:INITiate:CONTinuous

Description: Specifies whether the sweep/measurement is triggered continuously. If the value is set to ON or 1, then another sweep/measurement is triggered as soon as the current one is complete. If continuous is set to OFF or 0, then the instrument enters the "idle" state and waits for the `:INITiate[:IMMediate]` command or for `:INITiate:CONTinuous ON`. The default value is ON. That is, sending `:INIT:CONT` is equivalent to sending `:INIT:CONT ON`. The query version of the command returns a 1 if the instrument is continuously sweeping/measuring and returns a 0 if the instrument is in single sweep/measurement mode. Note that rapid toggling between ON and OFF is not allowed. The instrument must be allowed to make a full sweep before toggling can be done.

Syntax: `:INITiate:CONTinuous OFF|ON|0|1`

`:INITiate:CONTinuous?`

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: ON

Related Command: `:INITiate[:IMMediate]`
`:INITiate:HOLD`

Front Panel Access: **Shift-3** (Sweep), Sweep

Trigger Sweep/Measurement

:INITiate[:IMMediate]

Description: Initiates a sweep/measurement. If `:INITiate:CONTinuous` is set to ON, then this command is ignored. Use this command in combination with `:STATus:OPERation?` to synchronize the capture of one complete set of data. When this command is sent, the "sweep complete" bit of `:STATus:OPERation?` is set to 0, indicating that the measurement is not complete. The data collection is then triggered. The controlling program can poll `:STATus:OPERation?` to determine the status. When the "sweep complete" bit is set to 1, then data is ready to be retrieved.

When averaging is on, the sweep complete bit is set after the first sweep is completed. An `:INITiate[:IMMediate]` command must be issued for each additional sweep that is desired.

Syntax: `:INITiate[:IMMediate]`

Related Command: `:INITiate:CONTinuous`
`:STATus:OPERation?`

Front Panel Access: **Shift-3** (Sweep), Manual Trigger

Save On Crossing Limit

`:INITiate:SAVe:ON:EVENt:CROSsing:LIMit OFF|ON|0|1`

Description: When set to ON, automatically saves measurement data when the trace crosses a specified limit line. A limit line must be defined before this command can be enabled.

Default Value: OFF

Related Command: `:CALCulate:LIMit`
`:MMEMory:STORe:TRACe`

Front Panel Access: **Shift-7 (File)**, Save On Event, ...Crossing Limit

Save On Sweep Complete

`:INITiate:SAVe:ON:EVENt:SWEep OFF|ON|0|1`

Description: When set to ON, measurement data is automatically saved after each sweep is completed.

Default Value: OFF

Related Command: `:MMEMory:STORe:TRACe`

Front Panel Access: **Shift-7 (File)**, Save On Event, ...Sweep Complete

Save Then Stop

`:INITiate:SAVe:THEn:STOp OFF|ON|0|1`

Description: When set to ON, stops the sweep after a measurement is saved. If this setting is Off and Sweep Complete is On, a measurement is saved after every sweep.

Default Value: OFF

Related Command: `:MMEMory:STORe:TRACe`

Front Panel Access: **Shift-7 (File)**, Save On Event, Save Then Stop

5-9 :MEASure Subsystem

These commands take the instrument from its current state, enable the specified measurement and put the instrument into single sweep mode. They correct any parameters that are invalid given the new measurement state such that a valid measurement can take place. Other settings may be changed. Refer to the documentation of `:CONFigure` for each measurement. They then initiate the measurement. When the measurement is complete, they return the result.

To make a measurement with settings other than the "default" measurement settings applied by `:CONFigure`, do the following:

- Send the appropriate `:CONFigure` command to set the desired measurement.
- Modify the settings as required.
- Send the appropriate `:READ` command to measure and return the result.

To get the current measurement data, use the appropriate `:FETCh` command.

Measure Adjacent Channel Power Ratio

`:MEASure:ACPower?`

Description: Sets the active measurement to adjacent channel power ratio, sets the default measurement parameters, triggers a new measurement and returns the main channel power lower adjacent and upper adjacent channel power results. It is a combination of the commands `:CONFigure:ACPower; :READ:ACPower?` For a description of the default adjacent channel power ratio measurement parameters, refer to `:CONFigure:ACPower.` To make an adjacent channel power ratio measurement with settings other than the default values send:

```
:CONFigure:ACPower
Commands to set desired settings
:READ:ACPower?
```

Data is returned as 5 comma-separated values: main channel power, lower adjacent channel power, upper adjacent channel power, lower alternate channel power, upper alternate channel power.

Syntax: `:MEASure:ACPower?`

Cmd Parameter: **NA** (query only)

Query Response: **NA**

Default Unit: Current amplitude units

Related Command: `:CONFigure:ACPower`

Front Panel Access: **NA**

Measure Channel Power

:MEASure:CHPower:CHPower?

Description: Sets the active measurement to channel power, sets the default measurement parameters, triggers a new measurement and returns the channel power result. To measure both channel power and channel power density, use `:MEASure:CHPower?` It is a combination of the commands `:CONFigure:CHPower; :READ:CHPower:CHPower?` For a description of the default channel power measurement parameters, refer to `:CONFigure:CHPower.` To make a channel power measurement with settings other than the default values send:

```
:CONFigure:CHPower
Commands to set desired settings
:READ:CHPower:CHPower?
```

Syntax: `:MEASure:CHPower:CHPower?`

Cmd Parameter: **NA** (query only)

Query Response: **NA**

Default Unit: Current amplitude units

Related Command: `:MEASure:CHPower?`
`:MEASure:CHPower:DENSity?`
`:CONFigure:CHPower`

Front Panel Access: **NA**

Measure Channel Power Density

:MEASure:CHPower:DENSity?

Description: Sets the active measurement to channel power, sets the default measurement parameters, triggers a new measurement and returns the channel power density result. To measure both channel power and channel power density use `:MEASure:CHPower?` It is a combination of the commands `:CONFigure:CHPower; :READ:CHPower:DENSity?` For a description of the default channel power measurement parameters, refer to `:CONFigure:CHPower.` To make a channel power measurement with settings other than the default values send:

```
:CONFigure:CHPower
Commands to set desired settings
:READ:CHPower:DENSity?
```

Syntax: `:MEASure:CHPower:DENSity?`

Cmd Parameter: **NA** (query only)

Query Response: **NA**

Default Unit: Current amplitude units

Related Command:
```
:MEASure:CHPower?
:MEASure:CHPower:CHPower?
:CONFigure:CHPower
```

Front Panel Access: **NA**

Measure Channel Power/Density

:MEASure:CHPower?

Description: Sets the active measurement to channel power, sets the default measurement parameters, triggers a new measurement, and returns the channel power and channel power density results. It is a combination of the commands `:CONFigure:CHPower; :READ:CHPower?` For a description of the default channel power measurement parameters, refer to `:CONFigure:CHPower.` To make a channel power measurement with settings other than the default values send:

```
:CONFigure:CHPower
Commands to set desired settings
:READ:CHPower?
```

Data is returned as 2 comma-separated values: channel power, channel power density.

Syntax: `:MEASure:CHPower?`

Cmd Parameter: **NA** (query only)

Query Response: **NA**

Default Unit: Current amplitude units

Related Command: `:MEASure:CHPower:CHPower?`
`:MEASure:CHPower:DENSity?`
`:CONFigure:CHPower`

Front Panel Access: **NA**

Measure Occupied Bandwidth

`:MEASure:OBWidth?`

Description: Sets the active measurement to occupied bandwidth, sets the default measurement parameters, triggers a new measurement and returns the occupied bandwidth, percent of power and dB down results. It is a combination of the commands `:CONFigure:OBWidth;` `:READ:OBWidth?` For a description of the default occupied bandwidth measurement parameters, refer to `:CONFigure:OBWidth.` To make an occupied bandwidth measurement with settings other than the default values send:

```
:CONFigure:OBWidth
Commands to set desired settings
:READ:OBWidth?
```

Data is returned as 3 comma-separated values: occupied bandwidth, percent of power, dB down.

Syntax: `:MEASure:OBWidth?`

Cmd Parameter: **NA** (query only)

Query Response: **NA**

Default Unit: For OBW: Hz
For Percent of Power: %
For dB Down: dB

Related Command: `:CONFigure:OBWidth`
`:CONFigure:RF SPECtrum`

Front Panel Access: **NA**

5-10 :MMEMory Subsystem

The Mass MEMory subsystem contains functions that provide access to the instrument's setup and data storage.

Transfer Data

:MMEMory:DATA?

Title: Transfer Data

Description: Transfers the data stored in the given file from the instrument to the controlling program. Data is transferred in the form of `<header><block>`. The ASCII `<header>` specifies the number of data byes. It appears as #AX, where A is the number of digits in X, and X is the number of bytes in `<block>`. `<filename>` should be enclosed in either single quotes (' ') or double quotes (" ") and should contain a file extension (for example, `*.stp, *.jpg`), and the file must not be larger than 262136 bytes. Use the command `MMEMory:MSIS` to set the current storage location.

Syntax: `:MMEMory:DATA? <filename>`

Cmd Parameter: `<filename>`

Query Response: `<filename>`

Front Panel Access: **NA**

Delete Setup/Measurement

:MMEMory:DELete

Description: Removes a file specified by <filename> from the current mass storage device. <filename> should be enclosed in either single quotes (' ') or double quotes (" ") and it must include the file extension. Use the command MMEMory:MSIS to set the current file location.

Syntax: `:MMEMory:DELete <filename>`

Cmd Parameter: `<filename>`

Query Response: **NA** (no query)

Related Command: `:MMEMory:STORE:STATe`
`:MMEMory:STORe:TRACe`
`:MMEMory:MSIS INTernal|USB`

Front Panel Access: **Shift-7** (File), Delete, Delete Selected File

Recall Limit

:MMEMory:LOAD:LIMit

Description: Recalls a previously stored limit from the current storage location. The saved limit setting to be loaded is specified by `<filename>`. `<filename>` should be enclosed in either single quotes (' ') or double quotes (" ") and should contain a file extension ".`lim`". Note that the trace specified by `<filename>` should be available at the current mass storage location. Use the command `MMEMory:MSIS` to set the current mass storage location. File Extension: ".`lim`"

Syntax: `:MMEMory:LOAD:LIMit <filename>`

Cmd Parameter: `<filename>`

Query Response: **NA** (no query)

Example: To recall trace with file name "limit"

```
:MMEMory:LOAD:LIMit "limit.lim"
```

Related Command: `:MMEMory:STORe:LIMit`

Front Panel Access: **Shift-7** (File), Recall

Recall Setup

:MMEMory:LOAD:STATe

Description: Recalls a previously stored instrument setup in the current storage location. The setup file to be loaded is specified by `<filename>`. `<filename>` should be enclosed in either single quotes (' ') or double quotes (" ") and should contain a file extension ".`stp`". Use the command `MMEMory:MSIS` to set the current storage location. The `<integer>` parameter is not currently used, but it must be sent. Send a 1.

Note	When recalling setups for modes other than the current mode, Anritsu recommends that you first change modes, then recall the setup. Use ":INSTrument:NSELect" on page 7-3 or ":INSTrument[:SELect]" on page 7-4 to change the instrument state, then use `:MMEMory:LOAD:STATe` to recall the setup.

Syntax: `:MMEMory:LOAD:STATe <integer>,<filename>`

Cmd Parameter: `<integer>,<filename>`

Query Response: **NA** (no query)

Related Command: `:MMEMory:STORe:STATe`
`:MMEMory:MSIS INTernal|USB`

Front Panel Access: **Shift-7** (File), Recall

Recall Measurement

:MMEMory:LOAD:TRACe

Description: The instrument must be in the mode of the saved trace in order to recall that trace. Use `:INSTrument:SELect` or `:INSTrument:NSELect` to set the mode. Recalls a previously stored measurement trace from the current storage location. The saved measurement trace to be loaded is specified by `<filename>`. `<filename>` should be enclosed in either single quotes (‘ ’) or double quotes (“ ”) and should contain a file extension. Note that the trace specified by `<filename>` should be available at the current mass storage location. Use the command `MMEMory:MSIS` to set the current mass storage location. The `<integer>` parameter is not currently used, but it must be sent. Send a 1.

File name extensions:

“`.spa`” for SPA
`.vna`” for Cable & Antenna
“`.cdma`” for CDMA
“`.tds`” for TDSCDMA
“`.evdo`” for EVDO
“*.ia*” for Interference Analysis
“`.cs`” for Channel Scanner
“`.pm`” for Power Meter
“`.tm`” for Transmission Measurement
“.dmr2” for DMR 2 Analyzer measurements
“.p25” for P25 Analyzer measurements
“.nxdn” for NXDN Analyzer measurements
“.afp” for AMFMPM Analyzer measurements

Note: Extensions not available for T1 and Hi_PM.

Syntax: `:MMEMory:LOAD:TRACe <integer>,<filename>`

Cmd Parameter: `xx <integer>,<filename>`

Query Response: **NA** (no query)

Example: To recall trace with file name “trace”:

```
:MMEMory:LOAD:TRACe 1,”trace.spa”
```

Related Command: `:MMEMory:STORe:TRACe`
`:MMEMory:MSIS INTernal|USB`

Front Panel Access: **Shift-7** (File), Recall Measurement

Save Limit

:MMEMory:STORe:LIMit

Description: Stores the current limit setup into the file specified by `<filename>`. `<filename>` should be enclosed in either single quotes (‘ ’) or double quotes (“ ”) and should not contain a file extension. Use the command `MMEMory:MSIS` to set the current storage location.

Syntax: `:MMEMory:STORe:LIMit <filename>`

Cmd Parameter: `<filename>`

Query Response: **NA** (no query)

Related Command: `:MMEMory:MSIS INTernal|USB`

Front Panel Access: **Shift-7** (File), Save

Save Setup

:MMEMory:STORe:STATe

Description: Stores the current setup into the file specified by `<filename>`. `<filename>` should be enclosed in either single quotes (‘ ’) or double quotes (“ ”) and should not contain a file extension. Use the command `MMEMory:MSIS` to set the current storage location. The `<integer>` parameter is not currently used, but it must be sent. Send a value of 0.

Syntax: `:MMEMory:STORe:STATe <integer>,<filename>`

Cmd Parameter: `<integer>,<filename>`

Query Response: **NA** (no query)

Related Command: `:MMEMory:LOAD:STATe`
`:MMEMory:MSIS INTernal|USB`

Front Panel Access: **Shift-7** (File)

Save Measurement

:MMEMory:STORe:TRACe

Description: Stores the trace into the file specified by `<filename>`. `<filename>` should be enclosed in either single quotes (‘ ’) or double quotes (“ ”) and should not contain a file extension. Use the command `MMEMory:MSIS` to set the current storage location. Note that existing files of the same name will not be overwritten. The `<integer>` parameter is not currently used, but it must be sent. Send a 0.

Syntax: `:MMEMory:STORe:TRACe <integer>,<filename>`

Cmd Parameter: `<integer>,<filename>`

Query Response: **NA** (no query)

Example: To save the trace into the file name “trace”:

```
:MMEMory:STORe:TRACe 0,"trace"
```

Related Command: `:MMEMory:LOAD:TRACe`
`:MMEMory:MSIS INTernal|USB`

Front Panel Access: **Shift-7** (File), Save
Shift-7 (File), Save Measurement

5-11 :READ Subsystem

This set of commands combines the `:ABORt, :INITiate` and `:FETCh` commands. It aborts any current triggering sequence and sets the trigger state to idle. It then initiates a new active measurement (in other words, begins the collection of new data). When the measurement is complete, it returns the result. These commands will not switch to another measurement.

To get the current measurement data, use the `:FETCh` command.

Read Adjacent Channel Power Ratio

`:READ:ACPower?`

Description: Triggers a new adjacent channel power ratio measurement and returns the results: main channel power, lower adjacent channel power, and upper adjacent channel power. It is a combination of the commands `:ABORT; :INITiate; :FETCh:ACPower?` The channel power measurement must be the active measurement (specified by the command `:CONFigure:ACPower`). The current measurement can be queried using the command `:CONFigure?`

Data is returned as 5 comma-separated values: main channel power, lower adjacent channel power, upper adjacent channel power, lower alternate channel power, upper alternate channel power.

Syntax: `:READ:ACPower?`

Cmd Parameter: **NA** (query only)

Query Response: **NA**

Related Command: `:READ:CHPower?`
`:CONFigure:ACPower`
`:CONFigure:CHPower`

Front Panel Access: **NA**

Read Channel Power Density

:READ:CHPower:DENSity?

Description: Triggers a new channel power measurement and returns the channel power density result. It is a combination of the commands `:ABORT; :INITiate; :FETCh:CHPower:DENSity?` It returns only the channel power density, not the channel power. Use the command `:READ:CHPower?` to get both channel power and channel power density. The channel power measurement must be the active measurement (specified by `:CONFigure:CHPower`). The current measurement can be queried using `:CONFigure?` command.

Syntax: `:READ:CHPower:DENSity?`

Cmd Parameter: **NA** (query only)

Query Response: **NA**

Default Unit: Current amplitude units

Related Command: `:READ:CHPower?`
`:READ:CHPower:CHPower?`
`:CONFigure:CHPower`

Front Panel Access: **NA**

Read Channel Power

:READ:CHPower?

Description: Triggers a new channel power measurement and returns the results. It is a combination of the commands `:ABORT; :INITiate; :FETCh:CHPower?` The channel power measurement must be active. The current measurement can be queried using `:CONFigure?`

Syntax: `:READ:CHPower?`

Cmd Parameter: **NA** (query only)

Query Response: **NA**

Default Unit: dBm

Front Panel Access: **NA**

Read Occupied Bandwidth

`:READ:OBWidth?`

Description: Triggers a new occupied bandwidth measurement and returns the results: occupied bandwidth, percent of power and dB down. It is a combination of the commands `:ABORT; :INITiate; :FETCh:OBWidth?` The occupied bandwidth measurement must be the active measurement (specified by `:CONFigure:OBWidth`). The current measurement can be queried using `:CONFigure?`

Data is returned as 3 comma-separated values: occupied bandwidth, percent of power, dB down.

Syntax: `:READ:OBWidth?`

Cmd Parameter: **NA** (query only)

Query Response: **NA**

Default Unit: For OBW: Hz
For Percent of Power: %
For dB Down: dB

Related Command: `:CONFigure:OBWidth`

Front Panel Access: **NA**

5-12 :TRACe Subsystem

This subsystem contains commands related to the transfer of trace data to and from the instrument.

Trace Copy

:TRACe:COPY

Description: Copies Trace A to either Trace B or Trace C. Copying Trace A to Trace B is equivalent to pressing the **Shift-5** (Trace), Trace B Operations, A→B on the front panel. This stores Trace A into Trace B and turns on Trace B if it was off. Copying Trace A to Trace C is equivalent to pressing the **Shift-5** (Trace), Trace C Operations, A→C on the front panel. This stores Trace A into Trace C and turns on Trace C if it was off.

Syntax: `:TRACe:COPY TRACE1,TRACE2|TRACE3`

Cmd Parameter: `<char> TRACE1,TRACE2|TRACE3`

Query Response: **NA** (no query)

Example: To copy Trace A to Trace B:

```
:TRACe:COPY TRACE1,TRACE2
```

To copy Trace A to Trace C:

```
:TRACe:COPY TRACE1,Trace3
```

Front Panel Access: **NA**

Trace Exchange

:TRACe:EXCHange TRACE2,TRACE3

Description: Swaps Trace B and Trace C.

Syntax: `:TRACe:EXCHange TRACE2,TRACE3`

Cmd Parameter: `<char> TRACE2,TRACE3`

Query Response: **NA** (no query)

Front Panel Access: **Shift-5** (Trace), Trace B Operations, B↔C
Shift-5 (Trace), Trace C Operations, B↔C

Trace Header Transfer

:TRACe:PREamble?

Description: Returns trace header information for the specified trace. Data can be transferred to and from the 3 available display traces. Use the commands in the :MMEMory subsystem to store and recall traces from the instrument memory. The response begins with an ASCII header. The header specifies the number of following bytes. It appears as #AX, where A is the number of digits in X, and X is the number of bytes that follow the header. (Refer to "Example Response Format:" on page 3-106 for an example of the header.)

Parameters are returned in comma-delimited ASCII format. Each parameter is returned as `"NAME=VALUE[UNITS],"` Note that the parameters that are returned depend on the firmware version and that this document does not cover all parameter values that are returned by the command. Refer to Table 5-2 for valid parameter names.

Syntax: `:TRACe:PREamble? {1|2|3}`

Cmd Parameter: **NA** (query only)

Query Response: `[1]|2|3`

Range: 1|2|3

Front Panel Access: **NA**

Trace Header Parameters

Table 5-2 describes parameters that can be returned by the `:TRACe:PREamble?` command.

Table 5-2. Available Parameters in Spectrum Analyzer Mode (Sheet 1 of 6)

Parameter Name	Description
SN	Instrument serial #
UNIT_NAME	Instrument name
DESCR	Trace name
DATE	Trace date/time
BASE_VER	Base FW version
APP_NAME	Application name
APP_VER	Application FW version
UNITS	Amplitude units
CENTER_FREQ	Center frequency
SPAN	Frequency span
FREQ_STEP	Frequency step size
RBW	Resolution bandwidth
RBW_TYPE	RBW coupling auto/manual
VBW	Video bandwidth
VBW_TYPE	VBW coupling auto/manual
RBW_VBW_RATIO	RBW/VBW ratio
SPAN_RBW_RATIO	Span/RBW ratio
INPUT_ATTEN	Input attenuation
ATTEN_TYPE	Attenuation coupling auto/manual
REFERENCE_LEVEL	Reference level
SCALE	Y-axis scale
PREAMP_SET	Preamp state
REF_LEVEL_OFFSET	Reference level offset
DETECTION	Detection type
TRACE_AVERAGE	Number of traces to average
SWEEP_TYPE	Single/continuous

Table 5-2. Available Parameters in Spectrum Analyzer Mode (Sheet 2 of 6)

Parameter Name	Description
CURRENT_SIGNAL	Current signal index
CURRENT_CHANNEL	Current signal channel
TRACE_MODE	Normal/Avg/Max
TRACE_STATUS	TRACE_A_VIEW_NOT_BLANK: 0x0000000000000001 TRACE_A_WRITE_NOT_HOLD: 0x0000000000000002 TRACE_A_DATA_VALID: 0x0000000000000004 TRACE_B_VIEW_NOT_BLANK: 0x0000000000010000 TRACE_B_WRITE_NOT_HOLD: 0x0000000000020000 TRACE_B_DATA_VALID: 0x0000000000040000 TRACE_C_VIEW_NOT_BLANK: 0x0000000100000000 TRACE_C_WRITE_NOT_HOLD: 0x0000000200000000 TRACE_C_DATA_VALID: 0x0000000400000000 TRACE_C_IS_B_MINUS_A_ON: 0x0000001000000000 TRACE_C_IS_A_MINUS_B_ON: 0x0000002000000000
TRACE_COUNT	Number of traces averaged
UI_DATA_POINTS	Number of display points
IMPEDANCE	Input impedance
REFERENCE_FREQUENCY	Reference freq
SET_SWEEP_TIME	Minimum sweep time setting
TRIGGER_TYPE	Trigger type

Table 5-2. Available Parameters in Spectrum Analyzer Mode (Sheet 3 of 6)

Parameter Name	Description
VIDEO_TRIGGER_LEVEL	Video trigger level
TRIGGER_POSITION	Trigger position as a percent of the display
PEAK_THRESHOLD	Marker peak search threshold
MARKER_TABLE	Marker table status
ACTIVE_ MEASUREMENT	Current measurement
ANTENNA	Antenna index
OCC_BW_METHOD	Occupied bandwidth method
OCC_BW_PERCENT	Occupied bandwidth % of power setting
OCC_BW_DBC	Occupied bandwidth dBc setting
OCC_BW_MEASURED_ DB	Occupied bandwidth measured dBc value
OCC_BW_MEASURED_ PERCENT	Occupied bandwidth measured % value
OCC_BW_VALUE	Measured occupied bandwidth
OCC_BW_LINE_ MARKER_INFO	Mask off 16 bits at a time to get the display point location of the 3 OBW display indicators
CH_PWR_WIDTH	Channel power integration bandwidth
CH_PWR_VALUE	Measured channel power
CH_PWR_DENSITY	Measured channel power density
CH_PWR_LINE_ MARKER_INFO	Mask off 16 bits at a time to get the display point location of the 2 channel power display indicators
ACPR_MAIN_CH_BW	ACPR main channel bandwidth
ACPR_ADJC_CH_BW	ACPR adjacent channel bandwidth
ACPR_CHANNEL_ SPACING	ACPR channel spacing
ACPR_MAIN_CH_PWR	ACPR measured main channel power
ACPR_UPPER_CH_PWR	ACPR measured upper channel power
ACPR_LOWER_CH_ PWR	ACPR measured lower channel power
ACPR_LOWER_CH_ LINE_MARKER_INFO	Mask off 16 bits at a time to get the display point location of the 2 ACPR lower channel display indicators

Table 5-2. Available Parameters in Spectrum Analyzer Mode (Sheet 4 of 6)

Parameter Name	Description
ACPR_MAIN_CH_LINE_ MARKER_INFO	Mask off 16 bits at a time to get the display point location of the 2 ACPR main channel display indicators
ACPR_UPPER_CH_ LINE_MARKER_INFO	Mask off 16 bits at a time to get the display point location of the 2 ACPR upper channel display indicators
AM_FM_DEMOD_VOL	AM/FM demod volume
AM_FM_DEMOD_ FREQUENCY	AM/FM demod freq
AM_FM_DEMOD_TYPE	AM/FM demod type
AM_FM_DEMOD_TIME	AM/FM demod time
AM_FM_LINE_ MARKER	Display point location of the demodulation frequency
BEAT_FREQUENCY_ OSC_FREQUENCY	BFO oscillator freq
CI_C_TYPE	C/I measurement carrier type
CI_C_VALUE	C/I measurement measured carrier power
CI_I_BB_VALUE	C/I measurement measured broadband interference power
CI_I_NB_VALUE	C/I measurement measured narrowband interference power
CI_I_WB_VALUE	C/I measurement measured wideband interference power
CI_BB_VALUE	C/I measurement with broadband interference
CI_NB_VALUE	C/I measurement with narrowband interference
CI_WB_VALUE	C/I measurement with wideband interference
MKR_SPA_FREQNx	Marker x frequency (where x is the marker number 0-11, 0 represent the reference marker #1 and 1 represent delta marker #1, 2 represent reference marker #2, and 3 represent delta marker #2, and so on)
MKR_SPA_POINTx	Reference marker x display point
MKR_SPA_MAGNTx	Reference marker x magnitude
MKR_SPA_PRCNTx	Reference marker x display percentage

Table 5-2. Available Parameters in Spectrum Analyzer Mode (Sheet 5 of 6)

Parameter Name	Description
MKR_SPA_FLAGSx	Reference marker x flags: SPA_MKR_FLAG_ON_OFF: 0x00000001 SPA_MKR_FLAG_DELTA_MKR: 0x00000002 SPA_MKR_FLAG_SELECTED: 0x00000004 SPA_MKR_FLAG_DATA_INVALID: 0x00000008 SPA_MKR_FLAG_DATA_STALE: 0x00000010 SPA_MKR_FLAG_FIXED: 0x00000020 SPA_MKR_FLAG_MASK: 0x000000FF SPA_MKR_FLAG_DISPL_AMPL_HZ: 0x00000100 SPA_MKR_FLAG_DISPL_AMPL_PER_HZ: 0x00000200 SPA_MKR_FLAG_DISP_FLAG: 0x00000F00 SPA_MKR_FLAG_RELATIVE: 0x00001000 SPA_MKR_STANDARD: 0x10000000 SPA_MKR_FIELD_STRENGHT: 0x20000000 SPA_MKR_NOISE: 0x30000000 SPA_MKR_COUNTER: 0x40000000 SPA_MKR_TIME: 0x50000000

Table 5-2. Available Parameters in Spectrum Analyzer Mode (Sheet 6 of 6)

Parameter Name	Description
MKR_SPA_REF_TOx	Specifies which marker is the marker x reference to
MKR_SPA_TRACex	Specifies which trace the marker x is for.
LIM_LFLAGS_UP	Upper limit flags: LIMIT_FLAG_ON: 0x00000004 LIMIT_FLAG_ALARM_ON: 0x00000002
LIM_FREQNC_UPx	Upper limit point x freq (where x is the limit point number starting with 0)
LIM_MAGNTD_UPx	Upper limit point x amplitude
LIM_LFLAGS_LO	Lower limit flags: LIMIT_FLAG_ON: 0x00000004 LIMIT_FLAG_ALARM_ON: 0x00000002
LIM_FREQNC_LOx	Lower limit point x freq (where x is the limit point number starting with 0)
LIM_MAGNTD_LOx	Lower limit point x amplitude

Trace Data Transfer

:TRACe[:DATA]

Description: This command transfers data from the controlling program to the instrument. The query form transfers trace data from the instrument to the controller. When transferred to the instrument, data is enclosed in parentheses as `(<header><block>)`, and when transferred from the instrument, data is formatted as `<header><block>`.

The ASCII header specifies the number of data bytes. It appears as #AX, where A is the number of digits in X, and X is the number of bytes in the `<block>`.

The format of the block data in the query form is specified by `:FORMat:DATA`. The block data in the command form is always sent in ASCII format. Data can be transferred to and from the 3 available display traces. Use the commands in the :MMEMory subsystem to store and recall traces from the instrument memory. The command form does not support setting all trace points to a single value. To do this, send the same value to each point. Trace setup information can be acquired by using `:TRACe[:DATA]:PREamble?`.

To acquire the data from Trace A in the instrument, send `:TRACe[:DATA]? 1`. A 551 point trace is returned as `#42204<block data>`. `<block>` data could be in either INTeger,32 or REAL,32 format. In both cases, each data point has 4 bytes. So, 4 bytes per point multiplied by 551 data points gives 2204 bytes in `<block>` data. This example assumes that `:FORMat:DATA INTeger,32` or `:FORMat:DATA REAL,32` has been sent to the instrument before the query command is sent.

The query command returns a #0 if data is invalid for the active trace.

Syntax: `:TRACe[:DATA] {1|2|3},(<header><block>)`

`:TRACe[:DATA]? {1|2|3}`

Cmd Parameter: `{1|2|3},(<header><block>)`

Query Response: `{1|2|3}`

Related Command: `:FORMat:DATA`
`:TRACe[:DATA]:PREamble?`

Front Panel Access: **NA**

Trace View State

:TRACe{1|2|3}:DISPlay[:STATe]

Description: Specifies whether the designated trace should be displayable (visible) or hidden. TRACe1 corresponds to Trace A, TRACe2 corresponds to Trace B, and TRACe3 corresponds to Trace C. Setting the value to ON or to 1 sets the designated trace to be visible. Setting the value to OFF or to 0 sets the designated trace to be hidden. Note that issuing this command also sets the specified trace as the active trace.

Syntax:

```
:TRACe{1|2|3}:DISPlay[:STATe] OFF|ON|0|1
:TRACe{1|2|3}:DISPlay[:STATe]?
```

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: ON for Trace A
OFF for Trace B
OFF for Trace C

Example: To set Trace A to Blank:

```
:TRACe:DISPlay OFF
:TRACe1:DISPlay 0
```

To set Trace B to View:

```
:TRACe2:DISPlay ON
:TRACe2:DISPlay:STATe ON
:TRACe2:DISPlay 1
```

Front Panel Access: **Shift-5** (Trace), Trace [A|B|C], View|Blank

Trace Write State

:TRACe{1|2|3}:WRITe[:STATe]

Description: Specifies whether the designated trace state should be set to write or to hold. TRACe1 corresponds to Trace A, TRACe2 corresponds to Trace B, and TRACe3 corresponds to Trace C. Setting the state to ON or to 1 sets the specified trace to write. Setting the state to OFF or to 0 sets the specified trace to hold.

Syntax:
```
:TRACe{1|2|3}:WRITe[:STATe] OFF|ON|0|1
:TRACe{1|2|3}:WRITe[:STATe]?
```

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: ON for Trace A
OFF for Trace B
OFF for Trace C

Example: To set Trace A to Hold:

```
:TRACe:WRITe:STATe OFF
:TRACe:WRITe OFF
:TRACe:WRITe 0
```

To set Trace C to Write:

```
:TRACe3:WRITe ON
:TRACe3:WRITe:STATe ON
:TRACe3:WRITe 1
```

Related Command: `:TRACe:DATA`

Front Panel Access: **Shift-5** (Trace), Write|Hold

Trace A Operation

:TRACe1:OPERation

Description: This command specifies how successive traces are combined to produce the resulting display values.

Setting the operation to `NORMal` is equivalent to pressing **Shift-5** (Trace), Trace A Operations, Normal→A on the front panel. This displays a trace based on the detection method selected.

Setting the operation to `MAXHold` is equivalent to pressing **Shift-5** (Trace), Trace A Operations, Max Hold→A on the front panel. This displays the largest signal for each display point over multiple sweeps.

Setting the operation to `MINHold` is equivalent to pressing **Shift-5** (Trace), Trace A Operations, Min Hold→A on the front panel. This displays the smallest signal for each display point over multiple sweeps.

Setting the operation to `AVERage` is equivalent to pressing **Shift-5** (Trace), Trace A Operations, Average→A on the front panel. This displays the average value of multiple sweeps for each display point.

The query version of the command returns the current operation mode or `"NONE"` if no operation is set.

Syntax: `:TRACe1:OPERation NORMal|MAXHold|MINHold|AVERage`

`:TRACe1:OPERation?`

Cmd Parameter: `<char> NORMal|MAXHold|MINHold|AVERage`

Query Response: `<char> NORM|MAXH|MINH|AVER`

Default Value: NORMal

Related Command: `[:SENSe]:AVERage:TYPE`

Front Panel Access: **Shift-5** (Trace), Trace A Operations, Normal→A|Max Hold→A|Min Hold→A|Average→A

Trace B Operation

:TRACe2:OPERation

Description: This command specifies how successive traces are combined to produce the resulting display values.

Setting the operation to `MAXHold` is equivalent to pressing **Shift-5** (Trace), Trace B Operations, Max Hold→B on the front panel. This displays the largest signal for each display point over multiple sweeps.

Setting the operation to `MINHold` is equivalent to pressing **Shift-5** (Trace), Trace B Operations, Min Hold→B on the front panel. This displays the smallest signal for each display point over multiple sweeps.

The query version of the command returns the current operation mode or `"NONE"` if no operation is set.

Syntax: `:TRACe2:OPERation MAXHold|MINHold`

`:TRACe2:OPERation?`

Cmd Parameter: `<char> MAXHold|MINHold`

Query Response: `<char> MAXH|MINH`

Default Value: None

Range: MAXHold | MINHold

Front Panel Access: **Shift-5** (Trace), Trace B Operations, Max Hold→B | Min Hold→B

Trace C Operation

:TRACe3:OPERation

Description: This command specifies how successive traces are combined to produce the resulting display values.

Setting the operation to `MAXHold` is equivalent to pressing **Shift-5** (Trace), Trace C Operations, Max Hold→C on the front panel. This displays the largest signal for each display point over multiple sweeps.

Setting the operation to `MINHold` is equivalent to pressing **Shift-5** (Trace), Trace C Operations, Min Hold→C on the front panel. This displays the smallest signal for each display point over multiple sweeps.

Setting the operation to `A-B` is equivalent to pressing **Shift-5** (Trace), Trace C Operations, A-B→C. This displays the difference between Trace A and Trace B values in Trace C.

Setting the operation to `B-A` is equivalent to pressing **Shift-5** (Trace), Trace C Operations, B-A→C. This displays the difference between Trace B and Trace A values in Trace C.

The query version of the command returns the current operation mode or `"NONE"` if no operation is set.

Syntax: `:TRACe3:OPERation MAXHold|MINHold|A-B|B-A`

`:TRACe3:OPERation?`

Cmd Parameter: `<char> MAXHold|MINHold|A-B|B-A`

Query Response: `<char> MAXH|MINH|A-B|B-A`

Default Value: None

Range: MAXHold | MINHold | A-B | B-A

Front Panel Access: **Shift-5** (Trace), Trace C Operations, Max Hold→C | Min Hold→C | A-B→C | B-A→C

5-13 :TRIGger Subsystem

This subsystem contains commands related to the triggering of instrument functions for the purposes of synchronization. Related commands appear in the :ABORt and :INITiate subsystems.

Trigger Source

:TRIGger[:SEQuence]:SOURce

Description: This command defines the trigger source. `IMMediate` triggering is the equivalent of free-run triggering. `EXTernal` triggering is triggered when a TTL signal is applied to the External Trigger input connector. `EXTernal` triggering is always done on the rising edge of the signal. It is available only in zero span mode.

Syntax: `:TRIGger[:SEQuence]:SOURce IMMediate|EXTernal|VIDeo`

`:TRIGger[:SEQuence]:SOURce?`

Cmd Parameter: `<char> IMMediate|EXTernal|VIDeo`

Query Response: `<char> IMM|EXT|VID`

Default Value: Immediate

Range: IMMediate | EXTernal | VIDeo

Related Command: `:TRIGger[:SEQuence]:VIDeo:LEVel`
`:TRIGger[:SEQuence]:VIDeo:POSition`

Front Panel Access: **Shift-3** (Sweep), Trigger Type

Video Trigger Level

:TRIGger[:SEQuence]:VIDeo:LEVel

Description: This command sets the video triggering level.

Syntax: `:TRIGger[:SEQuence]:VIDeo:LEVel <amplitude>`

`:TRIGger[:SEQuence]:VIDeo:LEVel?`

Cmd Parameter: `<amplitude>`

Query Response: `<amplitude>`

Default Value: –65.0 dBm

Default Unit: Current amplitude unit

Range: +30 dBm to –130 dBm

Front Panel Access: **Shift-3** (Sweep), Trigger Type, Video

Video Trigger Position

:TRIGger[:SEQuence]:VIDeo:POSition

Description: This command sets the video triggering position as a percentage of the display. When the sweep is triggered, the trigger point is drawn at the location defined by the position. The graph appearing to the left of the trigger position is data acquired before the trigger. The graph appearing to the right of the trigger position is data acquired after the trigger.

Syntax: `:TRIGger[:SEQuence]:VIDeo:POSition <percentage>`

`:TRIGger[:SEQuence]:VIDeo:POSition?`

Cmd Parameter: `<percentage>`

Query Response: `<percentage>`

Default Value: 1

Default Unit: %

Range: 0% to 100%

Front Panel Access: **Shift-3** (Sweep), Trigger Type, Change Trigger Position

5-14 :UNIT Subsystem

The unit subsystem is used to modify the default units used for related parameters. These changes affect parameters in both commands and responses.

Measurement Units

:UNIT:POWer

Description: Sets the default amplitude units for input, output, and display.

Available units: dBm, dBV, dBmV, dBuV, V, W.

Note that linear units are not operational with SPA V3.06. The set command is non-operational with SPA V3.06 as well.

Syntax: `:UNIT:POWer DBM|DBV|DBMV|DBUV|V|W`

`:UNIT:POWer?`

Cmd Parameter: `<char> DBM|DBV|DBMV|DBUV|V|W`

Query Response: `<char> DBM|DBV|DBMV|DBUV|V|W`

Default Value: dBm

Front Panel Access: **Amplitude**, Units

5-15 [:SENSe] Subsystem

The commands in this subsystem relate to device-specific parameters, not signal-oriented parameters.

ACPR Adjacent Channel Bandwidth

[:SENSe]:ACPower:BANDwidth|BWIDth:ADJacent

Description: Sets the adjacent channel bandwidth for the ACPR measurement.

Syntax: `[:SENSe]:ACPower:BANDwidth|BWIDth:ADJacent <freq>`

`[:SENSe]:ACPower:BANDwidth|BWIDth:ADJacent?`

Cmd Parameter: `<freq>`

Query Response: `<freq>`

Default Value: 10.35 MHz

Default Unit: Hz

Front Panel Access: **Shift-4** (Measure), ACPR, Adj Ch BW

ACPR Main Channel Bandwidth

[:SENSe]:ACPower:BANDwidth|BWIDth:MAIN

Description: Sets the main channel bandwidth for the ACPR measurement.

Syntax: `[:SENSe]:ACPower:BANDwidth|BWIDth:MAIN <freq>`

`[:SENSe]:ACPower:BANDwidth|BWIDth:MAIN?`

Cmd Parameter: `<freq>`

Query Response: `<freq>`

Default Value: 10.35 MHz

Default Unit: Hz

Front Panel Access: **Shift-4** (Measure), ACPR, Main Ch BW

ACPR Channel Spacing

[:SENSe]:ACPower:BANDwidth|BWIDth:SPACing

Description: Sets the channel spacing for the ACPR measurement.

Syntax: `[:SENSe]:ACPower:BANDwidth|BWIDth:SPACing <freq>`

`[:SENSe]:ACPower:BANDwidth|BWIDth:SPACing?`

Cmd Parameter: `<freq>`

Query Response: `<freq>`

Default Value: 10.35 MHz

Default Unit: Hz

Front Panel Access: **Shift-4** (Measure), ACPR, Ch Spacing

ACPR Measurement State

[:SENSe]:ACPower:STATe

Description: Sets the state of the adjacent channel power ratio measurement, ON or OFF. When using `:CONFigure:ACPower`, the state is automatically set to ON.

Syntax: `[:SENSe]:ACPower:STATe OFF|ON|0|1`

`[:SENSe]:ACPower:STATe?`

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: OFF

Front Panel Access: **NA**

Number of Traces to Average

[:SENSe]:AVERage:COUNt

Description: Sets the number of traces to average.

Syntax: `[:SENSe]:AVERage:COUNt <integer>`

`[:SENSe]:AVERage:COUNt?`

Cmd Parameter: `<integer>,<integer>`

Query Response: `<integer>,<integer>`

Default Value: 10

Range: 2 to 65535

Front Panel Access: **Shift-5** (Trace), Trace A Operations, # of Averages

Trace Mode (Normal/Average/Max Hold/Min Hold)

[:SENSe]:AVERage:TYPE

Description: Specifies how successive traces are combined to produce the resulting display value. Setting the TYPE to `NONE` is the equivalent of setting the trace mode to "Normal→A" on the front panel. The displayed value for a point is the current measured value for that point. Setting the TYPE to `SCALar` is the equivalent of setting the trace mode to "Average→A" on the front panel. The displayed value for a point is the average of the last <integer> measured values where `<integer>` is set by `[:SENSe]:AVERage:COUNt`. Setting the TYPE to `MAXimum` is the equivalent of setting the trace mode to "Max Hold→A" on the front panel. The displayed value for a point is the maximum measured value for that point over sweeps. Setting the TYPE to `MINimum` is the equivalent of setting the trace mode to "Min Hold→A" on the front panel. The displayed value for a point is the minimum measured value for that point over sweeps.

Syntax: `[:SENSe]:AVERage:TYPE NONE|SCALar|MAXimum|MINimum`

`[:SENSe]:AVERage:TYPE?`

Cmd Parameter: `<char> NONE|SCALar|MAXimum|MINimum`

Query Response: `<char> NONE|SCAL|MAX|MIN`

Default Value: NONE

Example: To set the TYPE to SCALar:

```
:SENSe:AVERage:TYPE SCALar
```

To set the TYPE to MAXimum:

```
:SENSe:AVERage:TYPE MAXimum
```

Related Command: `[:SENSe]:AVERage:COUNt`

Front Panel Access: Shift-5 (Trace), Trace A Operations

Resolution Bandwidth

[:SENSe]:BANDwidth|BWIDth[:RESolution]

Description: Sets the resolution bandwidth. Note that using this command turns the automatic resolution bandwidth setting OFF.

Syntax: `[:SENSe]:BANDwidth|BWIDth[:RESolution] <freq>`

`[:SENSe]:BANDwidth|BWIDth[:RESolution]?`

Cmd Parameter: `<freq>`

Query Response: `<freq>`

Default Value: 3 MHz

Default Unit: Hz

Range: 10 Hz to 3 MHz in a 1:3 sequence

Related Command: `[:SENSe]:BANDwidth|BWIDth[:RESolution]:AUTO`

Front Panel Access: **BW**, RBW

Resolution Bandwidth Coupling

[:SENSe]:BANDwidth|BWIDth[:RESolution]:AUTO

Description: Sets the state of the coupling of the resolution bandwidth to the frequency span. Setting the value to ON or to 1 results in the resolution bandwidth being coupled to the span. That is, when the span changes, the resolution bandwidth changes. Setting the value to OFF or to 0 results in the resolution bandwidth being uncoupled from the span. That is, changing the span does not change the resolution bandwidth. When this command is issued, the resolution bandwidth setting itself does not change, only the coupling is affected. The default value is ON. That is, sending `:SENS:BAND:RES:AUTO` is equivalent to sending `:SENS:BAND:RES:AUTO ON.`

Syntax: `[:SENSe]:BANDwidth|BWIDth[:RESolution]:AUTO OFF|ON|0|1`

`[:SENSe]:BANDwidth|BWIDth[:RESolution]:AUTO?`

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: ON

Related Command: `[:SENSE]:BANDwidth|BWIDth[:RESolution]:RATio`

Front Panel Access: **BW**, Auto RBW

Resolution Bandwidth to Span Ratio

[:SENSe]:BANDwidth|BWIDth[:RESolution]:RATio

Description: Sets the ratio of the resolution bandwidth to the span for use when the resolution-bandwidth-to-span coupling is enabled. Note that the front panel interface sets the inverse ratio: the span to the resolution bandwidth.

Syntax: `[:SENSe]:BANDwidth|BWIDth[:RESolution]:RATio <number>`

`[:SENSe]:BANDwidth|BWIDth[:RESolution]:RATio?`

Cmd Parameter: `<number>`

Query Response: `<number>`

Default Value: 0.00333

Range: 0.00001 to 1

Related Command: `[:SENSe]:BANDwidth|BWIDth[:RESolution]:AUTO`

Front Panel Access: **BW**, Span/RBW (note that this is the inverse ratio)

Video Bandwidth

[:SENSe]:BANDwidth|BWIDth:VIDeo

Description: Sets the video bandwidth. Note that using this command turns the automatic video bandwidth setting OFF.

Syntax: `[:SENSe]:BANDwidth|BWIDth:VIDeo <freq>`

`[:SENSe]:BANDwidth|BWIDth:VIDeo?`

Cmd Parameter: `<freq>`

Query Response: **NA** (no query)

Default Value: 1 MHz

Default Unit: Hz

Range: 1 Hz to 3 MHz in a 1:3 sequence

Related Command: `[:SENSe]:BANDwidth|BWIDth:VIDeo:AUTO`

Front Panel Access: **BW**, VBW

Video Bandwidth Coupling

[:SENSe]:BANDwidth|BWIDth:VIDeo:AUTO

Description: Sets the state of the coupling of the video bandwidth to the resolution bandwidth. Setting the value to ON or to 1 results in the video bandwidth being coupled to the resolution bandwidth. That is, when the resolution bandwidth changes, the video bandwidth changes. Setting the value to OFF or to 0 results in the video bandwidth being uncoupled from the resolution bandwidth. That is, changing the resolution bandwidth will no longer change the video bandwidth. When this command is issued, the video bandwidth setting itself does not change, only the coupling is affected. The default value is ON. That is, sending `:SENS:BAND:VID:AUTO` is equivalent to sending `:SENS:BAND:VID:AUTO ON.`

Syntax: `[:SENSe]:BANDwidth|BWIDth:VIDeo:AUTO OFF|ON|0|1`

`[:SENSe]:BANDwidth|BWIDth:VIDeo:AUTO?`

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: ON

Front Panel Access: **BW**, Auto VBW

Video Bandwidth to Resolution Bandwidth Ratio

[:SENSe]:BANDwidth|BWIDth:VIDeo:RATio

Description: Sets the ratio of the video bandwidth to the resolution bandwidth for use when the video-to-resolution bandwidth coupling is enabled. Note that the front panel interface sets the inverse ratio: the resolution bandwidth to the video bandwidth, which is an integer. In other words, if you send 0.35, then the display will show 2 not 2.857.

Syntax: `[:SENSe]:BANDwidth|BWIDth:VIDeo:RATio <number>`

`[:SENSe]:BANDwidth|BWIDth:VIDeo:RATio?`

Cmd Parameter: `<number>`

Query Response: `<number>`

Default Value: 0.33

Range: 0.00001 to 1

Related Command: `[:SENSe]:BANDwidth|BWIDth:VIDeo:AUTO`

Front Panel Access: **BW**, RBW/VBW (note that this is the inverse ratio)

Channel Power Integration Bandwidth

[:SENSe]:CHPower:BANDwidth|BWIDth:INTegration

Description: Sets the integration bandwidth for the channel power measurement. Integration bandwidth must be less than or equal to the frequency span.

Syntax: `[:SENSe]:CHPower:BANDwidth|BWIDth:INTegration <freq>`

`[:SENSe]:CHPower:BANDwidth|BWIDth:INTegration?`

Cmd Parameter: `<freq>`

Query Response: `<freq>`

Default Value: 10.35 MHz

Default Unit: Hz

Related Command: `[:SENSe]:FREQuency:SPAN`

Front Panel Access: **Shift-4** (Measure), Channel Power, Ch Pwr Width

Channel Power Measurement State

[:SENSe]:CHPower:STATe

Description: Sets the state of the channel power measurement, ON or OFF. When using `:CONFigure:CHPower`, the state is automatically set to ON.

Syntax: `[:SENSe]:CHPower:STATe OFF|ON|0|1`

`[:SENSe]:CHPower:STATe?`

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: OFF

Related Command: `:CONFigure:ACPower`

Front Panel Access: **Shift-4** (Measure), Channel Power, On/Off

Input Impedance

[:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude]

Description: Sets the input impedance that is used for amplitude correction and conversion between units (dBm versus dBV versus Volts, and so forth). If the value of `<integer>` is 50, then no correction is performed. If the value of `<integer>` is 75, then correction is based on Anritsu adapter 12N50-75B.

Syntax: `[:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude] <integer>`

`[:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude]?`

Cmd Parameter: `<integer>,<integer>`

Query Response: `<integer>,<integer>`

Default Value: 50

Range: 50 or 75, all other values are treated as described in command:

`[:SENSe]:CORRection:IMPedance[:INPut]:OFFSet`

Related Command: `[:SENSe]:BANDwidth|BWIDth:VIDeo:RATio`

Front Panel Access: **Shift-8** (System), Application Options, Impedance

Other Input Impedance Loss

[:SENSe]:CORRection:IMPedance[:INPut]:OFFSet

Description: Sets the value that is used for amplitude correction when the value set by `[:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude]` is something other than 50 or 75. This value is not applied if the impedance is set to either 50 or 75.

Syntax: `[:SENSe]:CORRection:IMPedance[:INPut]:OFFSet <rel ampl>`

`[:SENSe]:CORRection:IMPedance[:INPut]:OFFSet?`

Cmd Parameter: `<rel ampl>`

Query Response: `<rel ampl>`

Default Value: 0

Default Unit: dB

Range: 0 dB to 100 dB

Related Command: `[:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude]`

Front Panel Access: Shift-8 (System), Application Options, Impedance

Detection Type

[:SENSe]:DETector[:FUNCtion]

Description: Sets the detection method for calculating each display point. Each display point represents several measurements. The detection type determines how the display point is derived from its associated measurements. POSitive Peak detection displays the maximum value of the associated measurements. RMS detection displays the average power of the associated measurements. NEGative Peak detection displays the minimum value of the associated measurements. SAMPle detection displays the “middle” point of those measurements that are associated with a display point. For example, if 3 measurement frequencies are associated with a given display point, then sample detection displays the value at the frequency of the second measurement point.

Syntax:
```
[:SENSe]:DETector[:FUNCtion]
POSitive|RMS|NEGative|SAMPle
[:SENSe]:DETector[:FUNCtion]?
```

Cmd Parameter: `<char> POSitive|RMS|NEGative|SAMPle`

Query Response: `<char> POS|RMS|NEG|SAMP`

Default Value: (Positive) Peak

Front Panel Access: **Shift-3** (Sweep), Detection

Frequency Reference Status

[:SENSe]:EXTRefstatus?

Description: Returns 0 for internal or GPS reference. Returns 1 for external reference.

Front Panel Access: None

Center Frequency

[:SENSe]:FREQuency:CENTer

Description: Sets the center frequency. Note that changing the value of the center frequency changes the value of the coupled parameters: `Start Frequency` and `Stop Frequency`. It may also change the value of the span.

Syntax: `[:SENSe]:FREQuency:CENTer <freq>`

`[:SENSe]:FREQuency:CENTer?`

Cmd Parameter: `<NRf> <freq> (hertz)`

Query Response: `<NR3> <freq> (hertz)`

Default Value: 2000250000 Hz for MS2024B, MS2034B
3000002500 Hz for MS2025B, MS2035B

Default Unit: Hz

Range: 500 kHz to 4 GHz for MS2024B, MS2034B
500 kHz to 6 GHz for MS2025B, MS2035B

Front Panel Access: **Freq**, Center Freq

Channel Selection

[:SENSe]:FREQuency:SIGStandard:CHANnel

Description: Sets the channel number for the selected signal standard.

Parameter: <number>

Syntax: `[:SENSe]:FREQuency:SIGStandard:CHANnel <number>`

`[:SENSe]:FREQuency:SIGStandard:CHANnel?`

Cmd Parameter: `<number>`

Query Response: `<number>`

Front Panel Access: **Freq**, Channel

Signal Standard

[:SENSe]:FREQuency:SIGStandard:NAMe

Description: Selects the desired signal standard from the list. The `<string>` argument is the name of the desired signal standard as displayed in the instrument current signal standard list. The list can be seen on the instrument by pressing the Signal Standard soft key in the Freq menu. The list can also be downloaded remotely and viewed by using Anritsu Master Software Tools. For example, if the desired Signal Standard is P-GSM 900 - Uplink, then the value of the `<string>` argument would be "P-GSM 900 - Uplink".

The query form of this command returns the name of the currently-selected Signal Standard on the list.

Syntax: `[:SENSe]:FREQuency:SIGStandard:NAMe <string>`

`[:SENSe]:FREQuency:SIGStandard:NAMe?`

Cmd Parameter: `<string>`

Query Response: `<string>`

Front Panel Access: **Freq**, Signal Standard

Frequency Span

[:SENSe]:FREQuency:SPAN

Description: Sets the frequency span. Setting the value of `<freq>` to 0 Hz is the equivalent of setting the span mode to zero span. Note that changing the value of the frequency span changes the value of the coupled parameters Start Frequency and Stop Frequency, and may change the Center Frequency.

Syntax: `[:SENSe]:FREQuency:SPAN <freq>`

`[:SENSe]:FREQuency:SPAN?`

Cmd Parameter: `<freq>`

Query Response: `<freq>`

Default Unit: Hz

Front Panel Access: **NA**

Frequency Span – Full

[:SENSe]:FREQuency:SPAN:FULL

Description: Sets the frequency span to full span. Note that changing the value of the frequency span changes the value of the coupled parameters, Start Frequency and Stop Frequency, and may change the Center Frequency.

Syntax: `[:SENSe]:FREQuency:SPAN:FULL`

Cmd Parameter: **NA**

Query Response: **NA** (no query)

Front Panel Access: **Span**, Full Span

Frequency Span – Last

[:SENSe]:FREQuency:SPAN:PREVious

Description: Sets the frequency span to the previous span value. Note that changing the value of the frequency span changes the value of the coupled parameters, Start Frequency and Stop Frequency, and may change the Center Frequency.

Syntax: `[:SENSe]:FREQuency:SPAN:PREVious`

Cmd Parameter: **NA**

Query Response: **NA** (no query)

Default Unit: Hz

Front Panel Access: **Span**, Last Span

Start Frequency

[:SENSe]:FREQuency:STARt

Description: Sets the start frequency. Note that in the spectrum analyzer, changing the value of the start frequency changes the value of the coupled parameters, Center Frequency and Span.

Syntax: `[:SENSe]:FREQuency:STARt <freq>`

`[:SENSe]:FREQuency:STARt?`

Cmd Parameter: `<freq>`

Query Response: `<freq>`

Default Value: 0 Hz

Default Unit: Hz

Related Command: `[:SENSe]:FREQuency:STOP?`

Front Panel Access: **Freq**, Start Freq

Frequency Step

[:SENSe]:FREQuency:STEP[:INCRement]

Description: Sets the frequency step to the given frequency value.

Syntax: `[:SENSe]:FREQuency:STEP[:INCRement] <freq>`

`[:SENSe]:FREQuency:STEP[:INCRement]?`

Cmd Parameter: `<freq>`

Query Response: `<freq>`

Default Value: 1 MHz

Default Unit: Hz

Range: 1 Hz to 20 GHz

Front Panel Access: **Freq**, Freq Step

Stop Frequency

[:SENSe]:FREQuency:STOP

Description: Sets the stop frequency. Note that in the spectrum analyzer, changing the value of the stop frequency will change the value of the coupled parameters, Center Frequency and Span.

Syntax: `[:SENSe]:FREQuency:STOP <freq>`

`[:SENSe]:FREQuency:STOP?`

Cmd Parameter: `<freq>`

Query Response: `<freq>`

Default Unit: Hz

Front Panel Access: **NA**

Field Strength Antenna

[:SENSe]:FSTRength:ANTenna

Description: Selects an antenna from the antenna list to use for field strength measurement result calculations. The `<antenna>` argument is a 1-based index of the position of the desired antenna in the instrument current antenna list. The list can be seen on the instrument by choosing the Antenna soft key in the F Strength menu. For example, if the desired antenna were the third item on the antenna listing, then the value of the `<antenna>` argument would be 3.

The query form of this command returns the index of the currently-selected antenna.

Syntax: `[:SENSe]:FSTRength:ANTenna <antenna>`

`[:SENSe]:FSTRength:ANTenna?`

Cmd Parameter: `<antenna>`

Query Response: `<antenna>`

Default Value: 1

Related Command: `:CONFigure:FSTRength`

Front Panel Access: **Shift-4** (Measure), Field Strength, Antenna

Field Strength Measurement State

`[:SENSe]:FSTRength:STATe`

Description: Sets the state of the field strength measurement, ON or OFF. When using `:CONFigure:FSTRength`, the state is automatically set to ON.

Syntax: `[:SENSe]:FSTRength:STATe OFF|ON|0|1`

`[:SENSe]:FSTRength:STATe?`

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: OFF

Related Command: `:CONFigure:CHPower`

Front Panel Access: **Shift-4** (Measure), Field Strength, On/Off

Occupied Bandwidth Measurement Method

[:SENSe]:OBWidth:METHod

Description: Sets the method for calculating occupied bandwidth. XDB calculates the occupied bandwidth based on points that are a specified number of dB below the carrier. Issue command `[:SENSe]:OBWidth:XDB` to set the number of dB to be used. PERCent calculates the occupied bandwidth based on points a specified percentage of the carrier power below the carrier. Issue command `[:SENSe]:OBWidth:PERCent` to set the percentage to be used.

Syntax: `[:SENSe]:OBWidth:METHod XDB|PERCent`

`[:SENSe]:OBWidth:METHod?`

Cmd Parameter: `<char> XDB|PERCent`

Query Response: `<char> XDB|PERCent`

Default Value: PERCent

Related Command: `[:SENSe]:OBWidth:XDB` `[:SENSe]:OBWidth:PERCent`

Front Panel Access: **Shift-4** (Measure), OCC BW

Occupied Bandwidth Percent of Power

[:SENSe]:OBWidth:PERCent

Description: This command sets the percentage of carrier power that is used to measure the occupied bandwidth. This value is used in the measurement if `:SENSe:OBWidth:METHod` is set to PERCent.

Syntax: `[:SENSe]:OBWidth:PERCent <percentage>`

`[:SENSe]:OBWidth:PERCent?`

Cmd Parameter: `<percentage>`

Query Response: `<percentage>`

Default Value: 99

Default Unit: %

Range: 0% to 100%

Related Command: `[:SENSe]:OBWidth:METHod`

Front Panel Access: **Shift-4** (Measure), OCC BW, %

Occupied Bandwidth Measurement State

[:SENSe]:OBWidth:STATe

Description: Sets the state of the occupied bandwidth measurement, ON or OFF. When using `:CONFigure:OBWidth`, the state is automatically set to ON.

Syntax: `[:SENSe]:OBWidth:STATe OFF|ON|0|1`

`[:SENSe]:OBWidth:STATe?`

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: OFF

Related Command: `:CONFigure:OBWidth`

Front Panel Access: Shift-4 (Measure), OCC BW, On/Off

Occupied Bandwidth dB Down

[:SENSe]:OBWidth:XDB

Description: This command sets the number of dB below the carrier that is used to measure the occupied bandwidth. This value is used in the measurement if `:SENSe:OBWidth:METHod` is set to XDB.

Syntax: `[:SENSe]:OBWidth:XDB <rel ampl>`

`[:SENSe]:OBWidth:XDB?`

Cmd Parameter: `<rel ampl>`

Query Response: `<rel ampl>`

Default Value: 3 dBc

Default Unit: dBc

Range: 0 dBc to 100 dBc

Related Command: `[:SENSe]:OBWidth:METHod`

Front Panel Access: **Shift-4** (Measure), OCC BW, dBc

Input Attenuation

[:SENSe]:POWer[:RF]:ATTenuation

Description: Sets the input attenuation. Note that issuing this command sets the automatic input attenuation OFF.

Syntax: `[:SENSe]:POWer[:RF]:ATTenuation <rel ampl>`

`[:SENSe]:POWer[:RF]:ATTenuation?`

Cmd Parameter: `<rel ampl>`

Query Response: `<rel ampl>`

Default Value: 30 dB

Default Unit: dB

Range: 0 dB to 65 dB

Related Command: `[:SENSe]:POWer[:RF]:ATTenuation:AUTO`

Front Panel Access: **Amplitude**, Atten Lvl

Input Attenuation Coupling

[:SENSe]:POWer[:RF]:ATTenuation:AUTO

Description: Sets the input attenuation coupling. Setting the value to ON or to 1 results in the input attenuation being coupled to the reference level. Setting the value to OFF or to 0 results in the input attenuation being uncoupled from the reference level. That is, changing the reference level does not change the input attenuation. When this command is issued, the input attenuator setting itself does not change, only the coupling is affected. The default value is ON. That is, sending `:SENS:POW:ATT:AUTO` is equivalent to sending `:SENS:POW:ATT:AUTO ON.`

Syntax: `[:SENSe]:POWer[:RF]:ATTenuation:AUTO OFF|ON|0|1`

`[:SENSe]:POWer[:RF]:ATTenuation:AUTO?`

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: ON

Related Command: `[:SENSe]:POWer[:RF]:ATTenuation`

Front Panel Access: **Amplitude**, Auto Atten

Preamp State

[:SENSe]:POWer[:RF]:GAIN[:STATe]

Description: Sets the state of the preamplifier (preamp). Note that this may cause a change in the reference level or attenuation or both.

Syntax: `[:SENSe]:POWer[:RF]:GAIN[:STATe] OFF|ON|0|1`

`[:SENSe]:POWer[:RF]:GAIN[:STATe]?`

Cmd Parameter: `<boolean> OFF|ON|0|1`

Query Response: `<bNR1> 0|1`

Default Value: OFF

Front Panel Access: **Amplitude**, Pre Amp

Sweep Status

[:SENSe]:SWEep:STATus?

Description: Returns 1 when the sweep is complete. Returns 0 when the sweep is in progress.

Syntax: `[:SENSe]:SWEep:STATus?`

Front Panel Access: None

Minimum Sweep Time

[:SENSe]:SWEep:TIME

Description: Sets the value of the minimum sweep time parameter. The sweep is completed in the shortest time possible. To sweep as fast as possible, enter the minimum value that is allowed for the sweep time.

Syntax: `[:SENSe]:SWEep:TIME`

`[:SENSe]:SWEep:TIME?`

Default Value: 1 ms

Default Unit: Seconds

Range: 10 µs to 600000000 µs

Front Panel Access: **Shift-3** (Sweep), Min Sweep Time

Trace Count

[:SENSe]:SWEep:TRACe?

Description: Returns a string with two integers representing the current trace's average count and the total number of trace averages.

Front Panel Access: None

Chapter 6 — AM/FM/PM Commands

6-1 :CALCulate Subsystem

The commands in this subsystem process data that has been collected via the SENSe subsystem. Commands may require the instrument to be in the proper mode or set up to use the feature of the command. For example, Marker commands function in one of the spectrum modes, Summary commands require the Summary mode. Use the `:AFP:DEM:MODE` command to set the desired mode.

`:CALCulate:MARKer{1|2|3|4|5|6}[:STATe] OFF|ON|0|1`
`:CALCulate:MARKer{1|2|3|4|5|6}[:STATe]?`

Title: Marker State

Description: Sets the specified marker on/off.

Parameter: OFF|ON|0|1

Parameter Type: <boolean>

Default Value: OFF

Example: To turn off reference marker #1:

```
:CALCulate:MARKer1:STATe OFF
```

Front Panel Access: Marker, On/Off

`:CALCulate:MARKer{1|2|3|4|5|6}:DELTa[:STATe] OFF|ON|0|1`
`:CALCulate:MARKer{1|2|3|4|5|6}:DELTa[:STATe]?`

Title: Delta Marker State

Description: Sets the specified delta marker on or off.

Parameter: OFF|ON|0|1

Parameter Type: <boolean>

Default Value: OFF

Example: To turn on delta marker #3:

```
:CALCulate:MARKer3:DELTa ON
:CALCulate:MARKer3:DELTa 1
:CALCulate:MARKer3:DELTa:STATe ON
:CALCulate:MARKer3:DELTa:STATe 1
```

To turn off delta marker #6:

```
:CALCulate:MARKer6:DELTa OFF
:CALCulate:MARKer6:DELTa:STATe OFF
:CALCulate:MARKer6:DELTa:STATe 0
```

Front Panel Access: Marker, Delta

:CALCulate:MARKer{1|2|3|4|5|6}:DELTa:X <x-parameter>
:CALCulate:MARKer{1|2|3|4|5|6}:DELTa:X?

Title: Delta Marker X Value

Description: Sets the location of the delta marker on the x-axis at the specified location <x-parameter> plus the reference marker x-axis. <x-parameter> is defined in the current x-axis units. The query version of the command returns the location of the delta marker on the x-axis.

Parameter: <x-parameter>

Default Unit: Hz or seconds if in Audio Waveform.

Example: If both the reference and delta marker #1 is currently at 1 GHz on the x-axis, send the command below to set the delta marker #1 to 2 GHz on the x-axis:

```
:CALCulate:MARKer1:DELTa:X 1GHz
```

(In Audio Waveform) If both the reference and delta marker #1 is currently at 25 µs on the x-axis, send the command below to set the delta marker to 50µs on the x-axis:

```
:CALCulate:MARKer1:DELTa:X 25µs
```

Related Command: `:CALCulate:MARKer[1]|2|3|4|5|6:X`

Front Panel Access: Marker, Delta

:CALCulate:MARKer{1|2|3|4|5|6}:DELTa:Y?

Title: Delta Marker Read Y Value

Description: Reads the current absolute Y value for the specified delta marker. The units are the units of the y-axis. In RF spectrum view, the value is returned in dBm. In Audio Spectrum or Audio Waveform view, the value is returned in % for AM, Hz for FM and Radians for PM.

Default Unit: Current y-axis unit

:CALCulate:MARKer{1|2|3|4|5|6}[:SET]:CENTer

Title: Marker Frequency to Center

Description: In RF spectrum view, this command sets the center frequency equal to the frequency of the specified marker. Note that this will result in a change to the start and stop frequencies and may also result in a change to the span. Note that this command is not valid in Audio Spectrum, Audio Waveform and Summary view.

Front Panel Access: Marker, Marker Freq to Center

:CALCulate:MARKer{1|2|3|4|5|6}:MAXimum

Title: Marker (Maximum) Peak Search

Description: Puts the specified marker at the maximum amplitude in the trace.

Front Panel Access: Marker, Marker [1/2/3/4/5/6], Peak Search
Marker, Marker [1/2/3/4/5/6], More Peak Options, Peak Search

:CALCulate:MARKer{1|2|3|4|5|6}[:SET]:RLEVel

Title: Marker to Reference Level

Description: Sets the reference level equal to the measured amplitude of the specified marker. Note that this may result in a change to the input attenuation. Note that this command is not valid in Audio Spectrum, Audio Waveform and Summary view.

Front Panel Access: Marker, Marker to Ref Lvl

:CALCulate:MARKer{1|2|3|4|5|6}:X <x-parameter>
:CALCulate:MARKer{1|2|3|4|5|6}:X?

Title: Marker X Value

Description: Sets the location of the marker on the x-axis at the specified location. <x-parameter> is defined in the current x-axis units. The query version of the command returns the location of the marker on the x-axis. Note that the marker is snapped to the data point closest to the specified value. If the specified marker is not on it is set to on.

Parameter: <x-parameter>

Default Unit: Hz or seconds if in Audio Waveform.

Example: To set reference marker #2 to 5 hertz on the x-axis:

```
:CALCulate:MARKer2:X 5
:CALCulate:MARKer2:X 5Hz
```

To set reference marker #1 to 1.5 GHz on the x-axis:

```
:CALCulate:MARKer:X 1.5GHz
:CALCulate:MARKer1:X 1.5GHz
```

(In Audio Waveform) To set reference marker #3 to 1.5 milli-seconds on the x-axis:

```
:CALCulate:MARKer3:X .0015
:CALCulate:MARKer3:X 1.5ms
```

:CALCulate:MARKer{1|2|3|4|5|6}:Y?

Title: Marker Read Y Value

Description: In RF spectrum view, the value is returned in dBm. In Audio Spectrum or Audio Waveform view, the value is returned in % for AM, Hz for FM and Radians for PM.

Default Unit: Current y-axis unit

:CALCulate:MARKer:AOFF

Title: Turn All Markers Off

Description: Turns off all markers.

Front Panel Access: Marker, More, All Markers Off

:CALCulate:MARKer:TABLe[:STATe] OFF|ON
:CALCulate:MARKer:TABLe[:STATe]?

Title: Marker Table State

Description: Turns the Marker Table on or off. Setting the value to ON will turn on the marker table. Setting the value to OFF will turn off the marker table.

Parameter: OFF|ON

Parameter Type: <boolean>

Default Value: OFF

Example: To turn on marker table:

```
:CALCulate:MARKer:TABLe ON
```

6-2 :DISPlay Subsystem

This subsystem provides commands that modify the display of data for the user. They do not modify the way in which data are returned to the controller.

:DISPlay:WINDow:TRACe:Y:ADJust

Title: Adjust Range

Description: Automatically adjusts reference level if input signal strength is too high (ADC error) or too low.

Example: `:DISPlay:WINDow:TRACe:Y:ADJust`

Front Panel Access: Amplitude, Adjust Range

:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision <rel ampl>
:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision?

Title: Scale

Description: Sets the scale (dB/division) for the y-axis in RF Spectrum view.

Parameter: <rel ampl>

Default Value: 10 dB/div

Default Unit: dB

Range: 1 dB to 15 dB

Front Panel Access: Amplitude, Scale

:DISPlay:WINDow:TRACe:Y:AFPanalyzer:PWR:OFFSet <rel ampl>
:DISPlay:WINDow:TRACe:Y:AFPanalyzer:PWR:OFFSet?

Title: Power Offset

Description: Sets the power offset value for the y-axis in RF Spectrum view.

Parameter: <rel ampl>

Default Value: 0 dB

Default Unit: dB

Range: -100 dB to 100 dB

Front Panel Access: Amplitude, Power Offset

6-3 :FORMat Subsystem

This subsystem contains commands that determine the formatting of numeric data when it is transferred. The format setting affects data in specific commands only. If a command is affected, it is noted in the command description.

:FORMat[:READings][:DATA] ASCii|INTeger,32|REAL,[<length>]
:FORMat[:READings][:DATA]?

Title: Numeric Data Format

Description: This command specifies the format in which data is returned in certain commands. The optional <length> parameter is needed for REAL format only. It defines the
length of the floating point number in bits. Valid values are 32 and 64. If no length is specified, the default length of REAL data is set to 64 bits.

ASCii format returns the data in comma-separated ASCII format. The units are
dBm for RF Spectrum, % for AM Audio Spectrum/Waveform, Hz for FM Audio Spectrum/Waveform, Radians for PM Audio Spectrum/Waveform.

This format requires many more bytes so it is the slowest format. INTeger,
32 values are signed 32-bit integers in little-endian byte order. This format
returns the data in 4-byte blocks. The units are mdBm for RF Spectrum, 1000*% for AM Audio Spectrum/Waveform, Hz for FM Audio Spectrum/Waveform, milli-Radians for PM Audio Spectrum/Waveform.

For example, if the measured result was -12.345 dBm, that value would be sent as -12345. REAL,32 values are 32-bit floating point numbers conforming to the IEEE 754 standard in little-endian byte order. This format returns the data in 4-byte binary format. The units are dBm for RF Spectrum, % for AM Audio Spectrum/Waveform,
Hz for FM Audio Spectrum/Waveform, Radians for PM Audio Spectrum/Waveform.

REAL,64 values are 64-bit floating point numbers conforming to the IEEE 754 standard in little-endian byte order. This format returns the data in 8-byte binary format. The units are dBm for RF Spectrum, % for AM Audio Spectrum/Waveform, Hz for FM Audio Spectrum/Waveform, Radians for PM Audio Spectrum/Waveform.

Both INTeger and REAL formats return a definite block length. Each transfer begins with an ASCII header such as #42204 for INTeger,32 and REAL,32 and #44408 for REAL,64. The first digit represents the number of following digits in the header (in this example, 4). The remainder of the header indicates the number of bytes that follow the header (in this example, 2204 for INT,32 and REAL,32 and 4408 for REAL,64). Divide the number of following bytes by the number of bytes in the data format chosen
(4 for both INTeger,32 and REAL,32, and 8 for REAL,64) to get the number of data points (in this example, 551).

Parameter: ASCii | INTeger,32 | REAL,[<length>]

Parameter Type: <char>

Default Value: ASCii

Related Command: `:TRACe[:DATA]`

6-4 :INITiate Subsystem

This subsystem controls the triggering of measurements.

:INITiate[:IMMediate]

Title: Trigger Sweep/Measurement

Description: Initiates a sweep/measurement. If :INITiate:CONTinuous is set to ON, this command is ignored. Use this command in combination with :STATus:OPERation? to synchronize the capture of one complete set of data. When this command is sent, the "sweep complete" bit of :STATus:OPERation? is set to 0, indicating that the measurement has not completed. The data collection is then triggered. The controlling program can poll :STATus:OPERation? to determine the status. When the "sweep complete" bit is set to 1, data is ready to be retrieved. An :INITiate[:IMMediate] command must be issued for each additional sweep desired.

Related Command: `:INITiate:CONTinuous`
`:STATus:OPERation?`

Front Panel Access: Shift-3 (Sweep), Manual Trigger

:INITiate:CONTinuous OFF|ON|0|1
:INITiate:CONTinuous?

Title: Continuous/Single Sweep

Description: Specifies whether the sweep/measurement is triggered continuously. If the value is set to ON or 1, another sweep/measurement is triggered as soon as the current one completes. If continuous is set to OFF or 0, the instrument enters the "idle" state and waits for the :INITiate[:IMMediate] command or for :INITiate:CONTinuous ON. The default value is ON. That is, sending :INIT:CONT is equivalent to sending :INIT:CONT ON. The query version of the command returns a 1 if the instrument is continuously sweeping/measuring and returns a 0 if the instrument is in single sweep/measurement mode. Note that rapid toggling between ON and OFF is not allowed. The instrument must be allowed to make a full sweep before toggling can be done.

Parameter: OFF|ON|0|1

Parameter Type: <boolean>

Default Value: ON

Related Command: `:INITiate[:IMMediate]`

Front Panel Access: Shift-3 (Sweep), Sweep

6-5 :MMEMory Subsystem

The Mass Memory subsystem contains functions that provide access to the instrument's setup and data storage.

:MMEMory:DELete <filename>

Title: Delete Setup/Measurement

Description: Removes a file specified by <filename> from the current mass storage device. <filename> should be enclosed in either single quotes (' ') or double quotes (" ") and it must include the file extension. Use the command MMEMory:MSIS to set the current file location.

Parameter: <filename>

Related Command:
```
:MMEMory:STORE:STATe
:MMEMory:STORe:TRACe
:MMEMory:MSIS INTernal|USB
```

Front Panel Access: Shift-7 (File), Delete, Delete Selected File

:MMEMory:LOAD:STATe <integer>,<filename>

Title: Recall Setup

Description: Recalls a previously stored instrument setup in the current save location. The setup file to be loaded is specified by <filename>. <filename> should be enclosed in either single quotes (' ') or double quotes (" ") and should contain a file extension ".stp". Use the command MMEMory:MSIS to set the current save location. The <integer> parameter is not currently used, but it must be sent. Send a 1.

Parameter: <integer>, <filename>

Related Command:
```
:MMEMory:STORe:STATe
:MMEMory:MSIS INTernal|USB
```

Front Panel Access: Shift-7 (File), Recall

:MMEMory:LOAD:TRACe <integer>,<filename>

Title: Recall Measurement

Description: The instrument must be in the mode of the saved trace in order to recall that trace.Use :INSTrument:SELect or :INSTrument:NSELect to set the mode. Recalls a previously stored measurement trace from the current save location. The saved measurement trace to be loaded is specified by <filename>. <filename> should be enclosed in either single quotes (' ') or double quotes (" ") and should contain a file extension. Note that the trace specified by <filename> should be available at the current save location.

Use the command MMEMory:MSIS to set the current save location. The <integer> parameter is not currently used, but it must be sent. Send a 1.

File name extensions:

"`.spa`" for SPA measurement
"`.mna`" for VNA and VVM measurements
"`.hipm`" for HiPM measurements
"`.pm`" for PM measurements
"`.cwsg`" for CWSG measurements
"`.afp`" for AM/FM/PM measurements
"`.ia`" for Interference Analysis measurements
"`.cs`" for Channel Scanner measurements
"`.wmxd`" for WiMAX
"`.wmxe`" for Mobile WiMAX
"`.lte`" for LTE measurements
"`.p25`" for P25 measurements
"`.p252`" for P25p2 measurements
"`.nxdn`" for NXDN measurements
"`.dpmr`" for dPMR measurements
"`.dmr2`" for DMR measurements
"`.ptc`" for PTC measurements
"`.tetra`" for TETRA measurements
"`.nbfm`" for NBFM measurements

Note: Extensions not available for T1 and Hi_PM.

Parameter: <integer>, <filename>

Example: To recall trace with file name "trace":

```
:MMEMory:LOAD:TRACe 1,"trace.afp"
```

Related Command:
```
:MMEMory:STORe:TRACe
:MMEMory:MSIS INTernal|USB
```

Front Panel Access: Shift-7 (File), Recall Measurement

:MMEMory:STORe:STATe <integer>,<filename>

Title: Save Setup

Description: Stores the current setup into the file specified by <filename>. <filename> should be enclosed in either single quotes (' ') or double quotes (" ") and should not contain a file extension. Use the command MMEMory:MSIS to set the current save location. The <integer> parameter is not currently used, but it must be sent. Send a value of 0.

Parameter: <integer>, <filename>

Related Command:
```
:MMEMory:LOAD:STATe
:MMEMory:MSIS INTernal|USB
```

Front Panel Access: Shift-7 (File)

:MMEMory:STORe:TRACe <integer>,<filename>

Title: Save Measurement

Description: Stores the trace into the file specified by <filename>. <filename> should be enclosed in either single quotes (‘ ’) or double quotes (“ ”) and should not contain a file extension. Use the command MMEMory:MSIS to set the current save location. The <integer> parameter is not currently used, but it must be sent. Send a 0. Note that existing files of the same name will not be overwritten.

Parameter: <integer>, <filename>

Example: To save the trace into the file name “trace”:

```
:MMEMory:STORe:TRACe 0,”trace”
```

Related Command:

```
:MMEMory:LOAD:TRACe
:MMEMory:MSIS INTernal|USB
```

Front Panel Access: Shift-7 (File), Save

6-6 :TRACe Subsystem

This subsystem contains commands related to the transfer of trace data to and from the instrument.

:TRACe[:DATA]?

Title: Trace Data Transfer

Description: This command transfers data from the controlling program to the instrument. The query form transfers trace data from the instrument to the controller. Data is transferred to the instrument enclosed in parentheses as (<header><block>) and from the instrument as <header><block>.

The ASCII header specifies the number of data bytes. It looks like #AX, where A is the number of digits in X and X is the number of bytes in the <block>. The format of the block data in the query form is specified by :FORMat:DATA. The block data in the command form is always sent in ASCII format.

To acquire the data from the trace in the instrument, send :TRACe[:DATA]? A 551 point trace is returned as #42204<block data>. <block> data could be in either INTeger,32 or REAL,32 format. In both cases, there is 4 bytes per data point. So, 4 bytes per point * 551 data points gives 2204 bytes in <block> data. This example assumes that :FORMat:DATA INTeger,32 or :FORMat:DATA REAL,32 has been sent to the instrument before the query command is sent.

The query command will return a #0 if data is invalid for the active trace.

Related Command: `:FORMat:DATA`

6-7 [:SENSe] Subsystem

The commands in this subsystem relate to device-specific parameters, not signal-oriented parameters.

[:SENSe]:AFPanalyzer:AVERage:COUNt <avg count>
[:SENSe]:AFPanalyzer:AVERage:COUNt?

Title: Measurement Average

Description: Sets the average count for the measurement data when in the summary mode (use `:AFP:DEM:MODE SUMM` to set summary mode). The query format of this command returns the value only in the summary mode. The query format returns nothing in other measurement modes.

Parameter: <number>

Range: 1 to 65535

Front Panel Access: Shift-4 (Measure), Average (access only in Summary mode)

[:SENSe]:AFPanalyzer:DEMod:DATA?

Title: Summary data

Description: This query returns the measurement values when in the summary mode (use `:AFP:DEM:MODE SUMM` to set summary mode). The order of the values are as follows:

<Modulation rate> in Hz, <RMS Deviation> in % or Hz or Rad depending on the demod type, <Peak-Peak/2 Dev> in % or Hz or Rad depending on the demod type, <SINAD> in dB, <THD> in % and <Distortion> in %. All values are comma separated.

Front Panel Access: Shift-4 (Measure), Audio Spectrum/Waveform

[:SENSe]:AFPanalyzer:DEMod:MODE RFSP|AFSP|AFWV|SUMMary
[:SENSe]:AFPanalyzer:DEMod:MODE?

Title: Demodulation Mode

Description: This command sets the demodulation mode (graph type) to RF Spectrum (RFSP), Audio Spectrum (AFSP), Audio Waveform (AFWV), or Summary (SUMM).

Parameter: RFSP|AFSP|AFWV|SUMMary

Default Value: RFSP

Example: To set the demodulation mode to Audio Waveform:

```
SENSe:AFPanalyzer:DEMod:MODE AFWV
```

Front Panel Access: Shift-4 (Measure)

[:SENSe]:AFPanalyzer:DEMod:TYPE AM|FM|PM
[:SENSe]:AFPanalyzer:DEMod:TYPE?

Title: Demodulation Type

Description: This command sets the demodulation type.

Parameter: AM | FM | PM

Default Value: AM

Example: To set the demodulation type to FM:

```
SENSe:AFPanalyzer:DEMod:TYPE FM
```

Front Panel Access: Setup, Demod Type

[:SENSe]:AFPanalyzer:FM:SCALe <percentage>
[:SENSe]:AFPanalyzer:FM:SCALe?

Title: FM Y-axis reference level percentage

Description: This command sets the Y-axis reference level as the percentage of the IF bandwidth while in the FM Audio spectrum/waveform display.

Parameter: <percentage>

Default Value: 50

Default Unit: %

Range: 0% to 100%

Front Panel Access: Shift-4 (Measure), Audio Spectrum/Waveform, Scale: % IFBW

[:SENSe]:AFPanalyzer:IFBW <freq>
[:SENSe]:AFPanalyzer:IFBW?

Title: IF bandwidth

Description: Sets the IF bandwidth. Note that using this command turns the automatic IF bandwidth setting OFF.

Parameter: <freq>

Default Value: 300 kHz

Default Unit: Hz

Range: 1 kHz to 300 kHz in a 1:3 sequence

Related Command: `:AFPanalyzer:IFBW:AUTO`

Front Panel Access: Setup, IFBW

[:SENSe]:AFPanalyzer:IFBW:AUTO 0|1
[:SENSe]:AFPanalyzer:IFBW:AUTO?

Title: IF bandwidth coupling

Description: Sets the state of the coupling of the IF bandwidth to the span. Setting the value to 1 will result in the IF bandwidth being coupled to the span. That is, when the span changes, the IF bandwidth changes. Setting the value to 0 will result in the IF bandwidth being un-coupled from the span. That is, changing the span will not change the IF bandwidth.

Parameter: 0 | 1

Parameter Type: <boolean>

Default Value: 1

Front Panel Access: Setup, Auto IFBW

[:SENSe]:AFPanalyzer:PM:SCALe <Radians>
[:SENSe]:AFPanalyzer:PM:SCALe?

Title: PM Y-axis reference level

Description: This command sets the Y-axis reference level while in the PM Audio spectrum/waveform display.

Parameter: <number>

Default Value: 3.140

Range: 3.140 to 3140

Front Panel Access: Shift-4 (Measure), Audio Spectrum/Waveform, Scale: milli-Rad

[:SENSe]:AFPanalyzer:RFSPectrum:DATA?

Title: Summary data

Description: This query returns the measurement values while in the RF spectrum mode (use `:AFP:DEM:MODE SUMM` to set summary mode). The order of the values are as follows:

<carrier power> in dBm, <carrier freq> in Hz, <Occ BW> in Hz. All values are comma separated.

Front Panel Access: Shift-4 (Measure), RF Spectrum.

[:SENSe]:AFPanalyzer:SPAN <freq>
[:SENSe]:AFPanalyzer:SPAN?

Title: Audio Frequency Span

Description: Sets the audio frequency span while in the Audio Spectrum mode (use `:AFP:DEM:MODE AFSP` to set audio spectrum mode). Valid values are 2 kHz, 5 kHz, 10 kHz, 20 kHz and 70 kHz.

Parameter: <freq>

Default Unit: Hz

Front Panel Access: Shift-4 (Measure), Audio Spectrum, Span

[:SENSe]:AFPanalyzer:SWEep:TIME <time>
[:SENSe]:AFPanalyzer:SWEep:TIME?

Title: Audio Waveform sweep time

Description: Sets the audio waveform sweep time while in the Audio Spectrum mode (use `:AFP:DEM:MODE AFSP` to set audio spectrum mode).

Parameter: <time>

Default Unit: Secs

Range: 50 µs to 50 ms.

Front Panel Access: Shift-4 (Measure), Audio Waveform, Sweep Time

[:SENSe]:AFPanalyzer:SUMMary:DATA?

Title: Summary data

Description: This query returns the measurement values in the Summary view (use `:AFP:DEM:MODE SUMM` to set summary mode). The order of the values are as follows:

<demod type> (AM | FM | PM), <RMS Deviation> in % or Hz or Rad depending on the demod type, <Peak+ Deviation> in % or Hz or Rad depending on the demod type, <Peak-Dev> in % or Hz or Rad depending on the demod type,
<Peak-Peak/2 Dev> in % or Hz or Rad depending on the demod type, <carrier power> in dBm, <carrier freq> in Hz, <Occ BW> in Hz, <Modulation rate> in Hz, <SINAD> in dB, <THD> in % and <Distortion> in %. All values are comma separated.

Front Panel Access: Shift-4 (Measure), Summary.

[:SENSe]:FREQuency:CENTer <freq>
[:SENSe]:FREQuency:CENTer?

Title: Center Frequency

Description: Sets the center frequency. Note that changing the value of the center frequency may also change the value of the span.

Parameter: <freq>

Default Unit: Hz

Front Panel Access: Freq, Center Freq

[:SENSe]:FREQuency:SIGStandard:CHANnel <number>
[:SENSe]:FREQuency:SIGStandard:CHANnel?

Title: Channel Selection

Description: Sets the channel number for the selected signal standard.

Parameter: <number>

Front Panel Access: Freq, Channel

[:SENSe]:FREQuency:SIGStandard:NAMe <string>
[:SENSe]:FREQuency:SIGStandard:NAMe?

Title: Signal Standard

Description: Selects the desired signal standard from the list. The <string> argument is the name of the desired signal standard as displayed in the instrument's current signal standard list. The list can be displayed on the instrument by choosing the Signal Standard submenu button in the Freq menu. The list can also be downloaded remotely and viewed using Anritsu Master Software Tools. For example, if the desired Signal Standard is:

P-GSM 900 - Uplink then the value of the <string> would be "P-GSM 900 - Uplink".

The query form of this command will return the name of the currently selected Signal Standard on the list.

Parameter: <string>

Front Panel Access: Freq, Signal Standard

[:SENSe]:FREQuency:SPAN <freq>
[:SENSe]:FREQuency:SPAN?

Title: Frequency Span

Description: Sets the frequency span. Minimum value and the maximum value are 10 kHz and 10 MHz respectively. Note that changing the value of the frequency span may change the Center Frequency.

Parameter: <freq>

Default Unit: Hz

Front Panel Access: Freq, Span

[:SENSe]:FREQuency:SPAN:FULL

Title: Frequency Span – Full

Description: Sets the frequency span to maximum span (10 MHz). Note that changing the value of the frequency span may change the Center Frequency.

Front Panel Access: Freq, Span, Max Span

[:SENSe]:FREQuency:SPAN:MINimum

Title: Frequency Span – Minimum

Description: Sets the frequency span to minimum span (10 kHz). Note that changing the value of the frequency span may change the Center Frequency.

Front Panel Access: Freq, Span, Min Span

[:SENSe]:FREQuency:SPAN:PREVious

Title: Frequency Span – Last

Description: Sets the frequency span to the previous span value. Note that changing the value of the frequency span may change the Center Frequency.

Default Unit: Hz

Front Panel Access: Freq, Span, Last Span

[:SENSe]:FREQuency:STEP[:INCRement] <freq>
[:SENSe]:FREQuency:STEP[:INCRement]?

Title: Frequency Step

Description: Sets the frequency step to the given frequency value.

Parameter: <freq>

Default Value: 1 MHz

Default Unit: Hz

Range: 1 Hz to 20 GHz

Front Panel Access: Freq, Freq Step

[:SENSe]:OBWidth:METHod XDB|PERCent
[:SENSe]:OBWidth:METHod?

Title: Occupied Bandwidth Measurement Method

Description: Sets the method for calculating occupied bandwidth. XDB calculates the occupied bandwidth based on points a specified number of dB below the carrier. Issue command [:SENSe]:OBWidth:XDB to set the number of dB to be used. PERCent calculates the occupied bandwidth based on points a specified percentage of the carrier power below the carrier. Issue command [:SENSe]:OBWidth:PERCent to set the percentage to be used.

Parameter: XDB|PERCent

Parameter Type: <char>

Default Value: PERCent

Related Command: `:OBWidth:XDB :OBWidth:PERCent`

Front Panel Access: Shift-4 (Measure), RF Spectrum, Occ BW Method

[:SENSe]:OBWidth:PERCent <percentage>
[:SENSe]:OBWidth:PERCent?

Title: Occupied Bandwidth Percent of Power

Description: This command sets the percentage of carrier power used to measure the occupied bandwidth. This value is used in the measurement if :SENSe:OBWidth:METHod is set to PERCent.

Parameter: <percentage>

Default Value: 99

Default Unit: %

Range: 0% to 100%

Related Command: `:OBWidth:METHod`

Front Panel Access: Shift-4 (Measure), RF Spectrum, %

[:SENSe]:OBWidth:XDB <rel ampl>
[:SENSe]:OBWidth:XDB?

Title: Occupied Bandwidth dB Down

Description: This command sets the number of dB below the carrier used to measure the occupied bandwidth. This value is used in the measurement if :SENSe:OBWidth:METHod is set to XDB.

Parameter: <rel ampl>

Default Value: 3 dBc

Default Unit: dBc

Range: 0 to 100 dBc

Related Command: `:OBWidth:METHod`

Front Panel Access: Shift-4 (Measure), RF Spectrum, dBc

Chapter 7 — All Mode Commands

7-1 Introduction

The commands that are listed in this chapter are functional in the following instrument modes of operation:

- Vector Network Analyzer
- Spectrum Analyzer
- Vector Voltmeter

Notational conventions are described in Section 2-10 “Command and Query Notational Conventions” on page 2-12.

7-2 All Mode Commands

Table 7-1. All Mode Commands Subsystems

Keyword	Parameter Data or Units
`:INSTrument`	Refer to “:INSTrument Subsystem” on page 7-2
`:MMEMory`	Refer to “:MMEMory Subsystem” on page 7-5
`:SYSTem`	Refer to “:SYSTem Subsystem” on page 7-11
`:SENSe:GPS`	Refer to “[:SENSe]:GPS Subsystem” on page 7-12
`:FETCh:GPS`	Refer to “:FETCh:GPS Subsystem” on page 7-14

7-3 :INSTrument Subsystem

One instrument may contain many logical instruments (“modes”). This subsystem controls the selection of the current instrument mode.

Query Available Modes

`:INSTrument:CATalog:FULL?`

Description: Returns a comma-separated list of available modes. Mode names are enclosed in double quotes (“ ”). The application number immediately follows the string name. For example: `“HI_PM”10,”MWVNA”26`

Syntax: `:INSTrument:CATalog:FULL?`

Cmd Parameter: **NA** (query only)

Query Response: **NA** (comma separated list)

Front Panel Access: **Shift-9** (Mode)

Select Mode by Number

:INSTrument:NSELect

Description: Sets the instrument mode based on the value of `<integer>`. The query version returns the number that is associated with the current mode. Use `:INSTrument:CATalog:FULL?` to get a list of available mode names and their integer representations.

Syntax:
```
:INSTrument:NSELect <integer>
:INSTrument:NSELect?
```

Cmd Parameter: `<NR1> (integer)`

```
  1 = SPA (Spectrum Analyzer mode)
 10 = HI_PM (High Accuracy Power Meter mode, Option 19)
 14 = IA (Interference Analysis, Option 25)
 15 = CS (Channel Scanner, Option 27)
 26 = MWVNA (Vector Network Analyzer mode)
 30 = AMFMPM (AM/FM/PM mode, Option 509)
102 = VVM (Vector Voltmeter mode, Option 15)
```

Query Response: `<NR1> (integer)`

```
  1 = SPA (Spectrum Analyzer mode)
 10 = HI_PM (High Accuracy Power Meter mode, Option 19)
 14 = IA (Interference Analysis, Option 25)
 15 = CS (Channel Scanner, Option 27)
 26 = MWVNA (Vector Network Analyzer mode)
 30 = AMFMPM (AM/FM/PM mode, Option 509)
102 = VVM (Vector Voltmeter mode, Option 15)
```

Related Command:
```
:INSTrument:CATalog:FULL?
:INSTrument[:SELect]
```

Front Panel Access: **Shift-9** (Mode)

Note	Switching modes can take longer than 60 seconds, depending on the application. Anritsu Company advises you to set the remote PC time-out to 120 seconds in order to avoid unexpected time-out errors.

Select Mode by Name

:INSTrument[:SELect]

Description: Sets the instrument mode based on the mode name that is specified by `<string>`. The query version returns the name of the current mode. Use `:INSTrument:CATalog:FULL?` to get a list of available modes. For example, for Vector Voltmeter, use "`VVM`".

Syntax:
```
:INSTrument[:SELect] <string>
:INSTrument[:SELect]?
```

Cmd Parameter:
```
<string> HI_PM|MWVNA|Power Monitor|VVM

AMFMPM = AM/FM/PM mode
CS = Channel Scanner
IA = Interference Analysis
HI_PM = High Accuracy Power Meter
MWVNA = Vector Network Analyzer
SPA = Spectrum Analyzer
VVM = Vector Voltmeter
```

Query Response:
```
<string> HI_PM|MWVNA|Power Monitor|VVM

AMFMPM = AM/FM/PM mode
CS = Channel Scanner
IA = Interference Analysis
HI_PM = High Accuracy Power Meter
MWVNA = Vector Network Analyzer
SPA = Spectrum Analyzer
VVM = Vector Voltmeter
```

Related Command:
```
:INSTrument:CATalog:FULL?
:INSTrument:NSELect
```

Front Panel Access: **Shift-9** (Mode)

Note	Switching modes can take longer than 60 seconds, depending on the application. Anritsu Company advises you to set the remote PC time-out to 120 seconds in order to avoid unexpected time-out errors.

7-4 :MMEMory Subsystem

The Mass MEMory subsystem contains functions that provide access to the instrument setup and data storage.

Table 7-2. :MMEMory Subsystem

Keyword	Parameter Form	Parameter Data or Units	Notes
`:MMEMory`			
`:MSIS?`		Refer to ":MMEMory:MSIS Subsystem" on page 7-8	
`:MSIS`		Refer to ":MMEMory:MSIS Subsystem" on page 7-8	
`:STORe`		Refer to ":MMEMory:STORe Subsystem" on page 7-10	

Transfer Data

:MMEMory:DATA?

Description: Transfers the data stored in the given file from the instrument to the controlling program. Data is transferred in the form of `<header><block>`. The ASCII `<header>` specifies the number of data bytes. It appears as `#AX`, where `A` is the number of digits in X, and `X` is the number of bytes in `<block>`.

`<filename>` should be enclosed in either single quotes (‘ ’) or double quotes (“ ”) and should contain a file extension (`.jpg`, for example). The file must not be larger than 524288 bytes. Use the command `:MMEMory:MSIS` to set the current storage location.

Syntax: `:MMEMory:DATA? <filename>`

Cmd Parameter: **NA** (query only)

Query Response: `<string> <filename>`

Related Command: `:MMEMory:MSIS INTernal|USB`

Front Panel Access: **NA**

Delete Data/Location

:MMEMory:DELete

Description: Removes a file specified by <filename> from the current mass storage device. <filename> should be enclosed in either single quotes (‘ ’) or double quotes (“ ”) and it must include the file extension. Use the command MMEMory:MSIS to set the current file location.

Syntax: `:MMEMory:DELete <filename>`

Cmd Parameter: `<string> <filename>`

Query Response: **NA** (no query)

Front Panel Access: **Shift-7** (File), Delete, Select or De-Select, Delete

Show Directory

MMEM:DIR?

Description: This command returns the non-recursive contents of the directory specified as the parameter. The parameter is case sensitive and must be enclosed in either single quotes (‘ ’) or double quotes (“ ”). Use “/” as a directory separator. For the internal memory, the parameter must start with "Internal".

The response is formatted as follows:

```
<file entry><sp>\r\n<file entry>
```

Where <file entry> is either:

```
<file name><sp><file type><sp><timestamp><sp>
<file size>
```

Or <file entry> is:

```
<directory name><sp><file type><sp><timestamp>
```

<*sp*> should be a single space.

<timestamp> is the number of seconds since Jan 1 1970.

Parameter: <directory>

Parameter Type: <string>

Front Panel Access: Shift-7 (File)

7-5 :MMEMory:MSIS Subsystem

The Mass MEMory "Mass Storage IS" subsystem contains commands for selecting a mass storage device that is used by all of the MMEMory commands.

Storage Location

:MMEMory:MSIS

Description: Sets the storage location. Setting the storage location to INTernal sets the current storage location to be the internal memory. Setting the storage location to USB sets the current storage location to be the USB Flash drive. Note that changing the current location to save files causes the "Copy to Destination" on the front panel to be automatically changed if the selected current location is the same as the destination. When the change occurs, the instrument cycles through each of the available storage locations.

Note that the storage location can be set independently and can be different for remote operation and front panel operation. Changing the copy location remotely does not change the location that is set and displayed on the front panel. Similarly, changing the copy location via the front panel does not affect the location that is used by the remote operation commands.

Note that the storage location must be available in order for it to be set.

Syntax: `:MMEMory:MSIS INTernal|USB`
`:MMEMory:MSIS?`

Cmd Parameter: `<char> INTernal|USB`

Query Response: `<char> INT|USB`

Related Command: `:MMEMory:MSIS:DESTination`

Front Panel Access: **NA**

Copy From Current Location To Destination

:MMEMory:MSIS:COPY

Description: Copies all measurements, setups, and `*.jpg` files that are stored in the current storage location to the "copy to destination" location.

Syntax: `:MMEMory:MSIS:COPY`

Cmd Parameter: **NA**

Query Response: **NA** (no query)

Related Command: `:MMEMory:MSIS`
`:MMEMory:MSIS:DESTination`

Front Panel Access: **NA**

Copy to Destination

:MMEMory:MSIS:DESTination

Description: Sets the destination to where measurements and setups in the current storage location are to be copied.

Setting the location to INTernal copies the files that are stored at the current storage location into the internal memory when the command `:MMEMory:MSIS:COPY` is sent.

Setting the location to USB copies the files that are stored at the current storage location into the USB Flash drive when the command `:MMEMory:MSIS:COPY` is sent.

Note that the storage location can be set independently and can be different for remote operation and front panel operation. Changing the save location remotely does not change the location that is set and displayed on the front panel. Similarly, changing the save location via the front panel does not affect the location that is used by the remote operation commands.

Syntax:
```
:MMEMory:MSIS:DESTination INTernal|USB
:MMEMory:MSIS:DESTination?
```

Cmd Parameter: `<char> INTernal|USB`

Query Response: `<char> INT|USB`

Related Command:
```
:MMEMory:MSIS
:MMEMory:MSIS:COPY
```

Front Panel Access: **Shift-7** (File), Copy, Scroll Destination Folder, Select or De-Select

7-6 :MMEMory:STORe Subsystem

The Mass MEMory STORe subsystem contains commands to transfer from the internal memory to the mass memory device.

Save Screen as JPEG

:MMEMory:STORe:JPEG

Description: Saves the current screen measurement as a JPEG file, which is specified by <file name> with the extension `*.jpg` to the current storage location. `<filename>` should be enclosed in either single quotes (‘ ’) or double quotes (“ ”) and should not contain a file extension. Use the command `:MMEMory:MSIS` to set the current storage location.

Syntax: `:MMEMory:STORe:JPEG <filename>`

Cmd Parameter: `<string> <filename>`

Query Response: **NA** (no query)

Example: To save the screen into the file named “trace”.

```
:MMEMory:STORe:JPEG "trace"
```

Related Command: `:MMEMory:DATA?`
`:MMEMory:MSIS:INTernal|USB`

Front Panel Access: **Shift-7** (File), Save

7-7 :SYSTem Subsystem

This subsystem contains commands that affect instrument functionality. This functionality does not directly relate to data collection, display, or transfer.

Query Installed Options

:SYSTem:OPTions?

Description: Returns a string of the installed options. Options are separated by a "/". The string returns "NONE" if no options are installed.

Syntax: `:SYSTem:OPTions?`

Cmd Parameter: **NA** (query only)

Query Response: **NA** (options are separated by "/" or "NONE")

Related Command: `*IDN?`

Preset

:SYSTem:PRESet

Description: This command restores all application parameters to their factory preset values. This command does not modify system parameters such as Ethernet configuration, language, volume, or brightness.
Note: After issuing this command, wait 30 seconds for the instrument to complete the preset condition before sending another command.

Syntax: `:SYSTem:PRESet`

Cmd Parameter: **NA**

Query Response: **NA** (no query)

Related Command: `*RST`

Front Panel Access: **Shift-1** (Preset), Preset

7-8 [:SENSe]:GPS Subsystem

This subsystem contains commands that relate to the optional GPS (Global Positioning System) on the instrument.

GPS On/Off

[:SENSe]:GPS

Description: Enables/disables optional GPS capability. The query version returns 0 when the GPS is Off and returns 1 when the GPS is On.

Requires Option 31.

Syntax: `[:SENSe]:GPS OFF|ON|0|1`
`[:SENSe]:GPS?`

Cmd Parameters: `<boolean> OFF|ON|0|1`

Query Responses: `<bNR1> 0|1`

Default Value: Off

Front Panel Access: **Shift 8** (System), GPS, GPS On/Off

GPS Reset

[:SENSe]:GPS:RESet

Description: Resets optional GPS receiver.

Requires Option 31.

Syntax: `[:SENSe]:GPS:RESet`

Cmd Parameters: **NA**

Query Responses: **NA** (no query)

Front Panel Access: **Shift 8** (System), GPS, Reset

GPS Antenna Current

[:SENSe]:GPS:CURRent?

Description: Query only. Reads the current draw of the GPS antenna in mA.

Requires Option 31.

Syntax: `[:SENSe]:GPS:CURRent?`

Cmd Parameters: **NA** (query only)

Query Responses: `<integer>`

Front Panel Access: **Shift 8** (System), GPS, GPS Info

GPS Antenna Voltage

[:SENSe]:GPS:VOLTage

Description: Sets and Reads the voltage setting for the GPS antenna. To set the voltage to 3.3 V, send the `0` parameter after the command. To set the voltage to 5 V, send the `1` parameter after the command. The query version returns `0` for an antenna voltage of 3.3 V and returns `1` for an antenna voltage of 5 V.

Requires Option 31.

Syntax:
```
[:SENSe]:GPS:VOLTage 0|1
[:SENSe]:GPS:VOLTage?
```

Cmd Parameters: `<bNR1> 0|1`

Query Responses: `<bNR1> 0|1`

Front Panel Access: **Shift 8** (System), GPS, GPS Voltage
Shift 8 (System), GPS, GPS Info

7-9 :FETCh:GPS Subsystem

The commands in this subsystem return the most recent measured GPS data.

Fetch GPS Fix Data

:FETCh:GPS?

Description: Returns the most recent GPS fix information from the optional GPS receiver.

The results are returned as a set of comma-delimited values in the following format:

`<fix status>, <date/time>, <latitude>, <longitude>`

The `<fix status>` field will be either `"GOOD FIX"` or `"NO FIX"` depending whether the GPS receiver is currently calculating position data. If `"NO FIX"` is the value of the `<fix status>` field, then no data follows.

The date and time (`<date/time>` field) are returned in the following format:

`Www Mmm dd hh:mm:ss yyyy`

Where `Www` is the weekday in letters, `Mmm` is the month in letters, `dd` is the day of the month, `hh:mm:ss` is the time (24-hour time), and `yyyy` is the year.

Both `<latitude>` and `<longitude>` fields are expressed in radians. A negative latitude value corresponds to a "south" reading. A negative longitude value corresponds to a "west" reading.

Requires Option 31.

Syntax: `:FETCh:GPS?`

Cmd Parameters: **NA** (query only)

Query Responses: `<string>, <arg>, <NR2>, <NR2> for parameter data of <fix status>, <date/time>, <latitude>, <longitude>`

Default Unit: Radians

Front Panel Access: NA

:SYSTem:WINDow:CLOSe

Title: Close active information window

Description: This command closes an active information window or a selection window.

Parameter: N/A

Parameter Type: N/A

Front Panel Access : Shift - Esc, Close Window

Appendix A — Example

A-1 Introduction

This appendix provides examples of C/C++ and Visual Basic coding.

A-2 C/C++

This example is run on the command line. It sends the *IDN? query to the instrument and prints the response to the console.

```
// IdnExample.cpp : Microsoft Visual Studio-Generated Example
//     Based on Example 2-1 in the NI-VISA User Manual
//     Usage : IdnExample "TCPIP::xxx.xxx.xxx.xxx::inst0::INSTR"
//        where xxx.xxx.xxx.xxx is the IP address of the
//        instrument.
//     Output : The string identity string returned from the
//        instrument.
//     VISA Header : visa.h (must be included)
//     VISA Library : visa32.lib (must be linked with)

#include "stdafx.h"
#include "stdio.h"
#include "string.h"
#include "visa.h"

#define BUFFER_SIZE 255

int main(int argc, char* argv[])
{
     ViStatus status; /* For checking errors */
     ViSession defaultRM, instr; /* Communication channels */
     ViUInt32 retCount; /* Return count from string I/O */
     ViChar buffer[BUFFER_SIZE]; /* Buffer for string I/O */
     char tempDisplay[BUFFER_SIZE]; /* Display buffer for example */
     char *pAddress;

     /* Make sure we got our address. */
     if ( argc < 2 )
     {
       printf("Usage: IdnExample
       \"TCPIP::xxx.xxx.xxx.xxx::inst0::INSTR\"\n");
       printf("\t where xxx.xxx.xxx.xxx is the IP address of your
       instrument.\n");
       return -1;
     }
```

```
/* Store the address. */
pAddress = argv[1];

/* Begin by initializing the system*/
status = viOpenDefaultRM(&defaultRM);

if (status < VI_SUCCESS)
{
     /* Error Initializing VISA...exiting*/
     printf(”Can't initialize VISA\n”);
     return -1;
}

/* Open communication with TCP/IP device at xxx.xxx.xxx.xxx*/
/* NOTE: For simplicity, we will not show error checking*/
/* TODO: Add error handling. */
status = viOpen(defaultRM, pAddress, VI_NULL, VI_NULL, &instr);

/* Set the timeout for message-based communication*/
/* TODO: Add error handling. */
status = viSetAttribute(instr, VI_ATTR_TMO_VALUE, 120000);

/* Ask the device for identification */
sprintf(buffer, “*IDN?\n”);
status = viWrite(instr, (unsigned char *)&buffer[0], 6, &retCount);
status = viRead(instr, (unsigned char *)buffer, BUFFER_SIZE,
&retCount);

/* TODO: Add code to process data. */
strncpy(tempDisplay, buffer, retCount);
tempDisplay[retCount] = 0;   /* Null-terminate display string. */
printf(”*IDN? Returned %d bytes: %s\n”, retCount, tempDisplay);
```

```
    /* Close down the system */
    /* TODO: Add error handling. */
    status = viClose(instr);
    status = viClose(defaultRM);

    return 0;
}
```

A-3 Visual Basic

This function can be called in a Visual Basic program. It sends the *IDN? query to the instrument and returns the byte count and ASCII response string.

```
Rem This example is based on Example 2-1 from the NI-VISA User Manual.

Public Sub IdnMain(ByVal address As String, ByRef byteCount As String,
ByRef returnBytes As String)
     Const BUFFER_SIZE = 200
     Dim stat As ViStatus
     Dim dfltRM As ViSession
     Dim sesn As ViSession
     Dim retCount As Long
     Dim buffer As String * BUFFER_SIZE

     Rem ***Include visa32.dll as a reference in your project.***

     Rem Begin by initializing the system
     stat = viOpenDefaultRM(dfltRM)
     If (stat < VI_SUCCESS) Then
       Rem Error initializing VISA...exiting
       MsgBox "Can't initialize VISA"
       Exit Sub
     End If

     Rem Open communication with Device
     Rem NOTE: For simplicity, we will not show error checking
     Rem TODO: Add error handling.
     stat = viOpen(dfltRM, address, VI_NULL, VI_NULL, sesn)
```

```
    Rem Set the timeout for message-based communication
    Rem TODO: Add error handling.
    stat = viSetAttribute(sesn, VI_ATTR_TMO_VALUE, 120000)

    Rem Ask the device for identification
    Rem TODO: Add error handling.
    stat = viWrite(sesn, "*IDN?", 5, retCount)
    stat = viRead(sesn, buffer, BUFFER_SIZE, retCount)

    Rem TODO: Add code to process the data.
    byteCount = retCount
    returnBytes = Left(buffer, retCount)

    Rem Close down the system
    Rem TODO: Add error handling.
    stat = viClose(sesn)
    stat = viClose(dfltRM)
End Sub
```

Appendix B — List of All SCPI Commands

:ABORt. . . . . SPA 5-2
:CALCulate:LIMit:ALARm. . . . . SPA 5-3
:CALCulate:LIMit:ALARm. . . . . VNA 3-6
:CALCulate:LIMit:CENTer . . . . . SPA 5-6
:CALCulate:LIMit:ENVelope:CREate . . . . . SPA 5-7
:CALCulate:LIMit:ENVelope:OFFSet . . . . . SPA 5-7
:CALCulate:LIMit:ENVelope:POINt . . . . . SPA 5-8
:CALCulate:LIMit:ENVelope:SHAPe . . . . . SPA 5-8
:CALCulate:LIMit:ENVelope:UPDate:X . . . . . SPA 5-9
:CALCulate:LIMit:ENVelope:UPDate:Y . . . . . SPA 5-9
:CALCulate:LIMit:FAIL? . . . . . SPA 5-9
:CALCulate:LIMit:LOWer:ALARm . . . . . SPA 5-10
:CALCulate:LIMit:LOWer:FAIL? . . . . . SPA 5-10
:CALCulate:LIMit:LOWer:POINt? . . . . . SPA 5-10
:CALCulate:LIMit:LOWer[:STATe] . . . . . SPA 5-11
:CALCulate:LIMit:LTYPe . . . . . SPA 5-12
:CALCulate:LIMit:MIRRor. . . . . SPA 5-12
:CALCulate:LIMit:POINt:ADD . . . . . SPA 5-12
:CALCulate:LIMit:POINt:ADD . . . . . VNA 3-14
:CALCulate:LIMit:POINt:DELete . . . . . SPA 5-13
:CALCulate:LIMit:POINt:DELete . . . . . VNA 3-14
:CALCulate:LIMit:POINt:LEFT . . . . . SPA 5-13
:CALCulate:LIMit:POINt:LEFT . . . . . VNA 3-15
:CALCulate:LIMit:POINt:RIGHt. . . . . SPA 5-13
:CALCulate:LIMit:POINt:RIGHt. . . . . VNA 3-15
:CALCulate:LIMit:POINt:X . . . . . SPA 5-14
:CALCulate:LIMit:POINt:X . . . . . VNA 3-16

:CALCulate:LIMit:POINt:Y . . . . . SPA 5-14
:CALCulate:LIMit:POINt:Y . . . . . VNA 3-17
:CALCulate:LIMit:POINt? . . . . . SPA 5-15
:CALCulate:LIMit:POINt? . . . . . VNA 3-13
:CALCulate:LIMit:TYPe . . . . . SPA 5-15
:CALCulate:LIMit:TYPE. . . . . VNA 3-19
:CALCulate:LIMit:UPPer:ALARm . . . . . SPA 5-16
:CALCulate:LIMit:UPPer:FAIL? . . . . . SPA 5-16
:CALCulate:LIMit:UPPer:POINt? . . . . . SPA 5-16
:CALCulate:LIMit:UPPer[:STATe]. . . . . SPA 5-17
:CALCulate:LIMit:VALue . . . . . SPA 5-17
:CALCulate:LIMit:VERTical:ADD . . . . . SPA 5-18
:CALCulate:LIMit:X . . . . . SPA 5-18
:CALCulate:LIMit:X . . . . . VNA 3-28
:CALCulate:LIMit:Y . . . . . SPA 5-19
:CALCulate:LIMit:Y . . . . . VNA 3-29
:CALCulate:LIMit[:SET]:DEFault . . . . . SPA 5-20
:CALCulate:LIMit[:STATe]. . . . . SPA 5-20
:CALCulate:LIMit[:STATe]. . . . . VNA 3-18
:CALCulate:MARKer:AOFF . . . . . SPA 5-21
:CALCulate:MARKer:AOFF . . . . . VNA 3-30
:CALCulate:MARKer:AOFF . . . . . AM/FM/PM 6-4
:CALCulate:MARKer:DATA? . . . . . VNA 3-30
:CALCulate:MARKer:DISPlay:FORMat . . . . . VNA 3-33
:CALCulate:MARKer:PEAK:THReshold . . . . . SPA 5-21
:CALCulate:MARKer:TABLe:DATA?. . . . . SPA 5-21
:CALCulate:MARKer:TABLe[:STATe] OFF|ON
:CALCulate:MARKer:TABLe[:STATe]? . . . . . AM/FM/PM 6-4
:CALCulate:MARKer:TABLe[:STATe] . . . . . SPA 5-22
:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:DELTa:REFerence .
VNA 3-31

:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:DELTa[:STATe] VNA 3-32

:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:MAXimum VNA 3-35

:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:MINimum VNA 3-35

:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:TYPE. . . . VNA 3-38

:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:X . . . . . VNA 3-39

:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:Y? . . . . . VNA 3-40

:CALCulate:MARKer{1|2|3|4|5|6}:DELTa:NOISe[:STATe] . . . . SPA 5-22

:CALCulate:MARKer{1|2|3|4|5|6}:DELTa:X <x-parameter>
:CALCulate:MARKer{1|2|3|4|5|6}:DELTa:X? . . . . . AM/FM/PM 6-2

:CALCulate:MARKer{1|2|3|4|5|6}:DELTa:X . . . . . SPA 5-23

:CALCulate:MARKer{1|2|3|4|5|6}:DELTa:Y? . . . . . SPA 5-23

:CALCulate:MARKer{1|2|3|4|5|6}:DELTa:Y? . . . . . AM/FM/PM 6-2

:CALCulate:MARKer{1|2|3|4|5|6}:DELTa[:SET]:SPAN . . . . . SPA 5-24

:CALCulate:MARKer{1|2|3|4|5|6}:DELTa[:STATe] OFF|ON|0|1
:CALCulate:MARKer{1|2|3|4|5|6}:DELTa[:STATe]?. . . . . AM/FM/PM 6-1

:CALCulate:MARKer{1|2|3|4|5|6}:DELTa[:STATe]. . . . . SPA 5-24

:CALCulate:MARKer{1|2|3|4|5|6}:FCOunt[:STATe] . . . . . SPA 5-25

:CALCulate:MARKer{1|2|3|4|5|6}:FIXed[:STATe]. . . . . SPA 5-25

:CALCulate:MARKer{1|2|3|4|5|6}:MAXimum . . . . . SPA 5-26

:CALCulate:MARKer{1|2|3|4|5|6}:MAXimum . . . . . AM/FM/PM 6-3

:CALCulate:MARKer{1|2|3|4|5|6}:MAXimum:LEFT. . . . . SPA 5-26

:CALCulate:MARKer{1|2|3|4|5|6}:MAXimum:NEXT . . . . . SPA 5-26

:CALCulate:MARKer{1|2|3|4|5|6}:MAXimum:RIGHt . . . . . SPA 5-27

:CALCulate:MARKer{1|2|3|4|5|6}:NOISe[:STATe] . . . . . SPA 5-27

:CALCulate:MARKer{1|2|3|4|5|6}:X <x-parameter>
:CALCulate:MARKer{1|2|3|4|5|6}:X? . . . . . AM/FM/PM 6-3

:CALCulate:MARKer{1|2|3|4|5|6}:X. . . . . SPA 5-28

:CALCulate:MARKer{1|2|3|4|5|6}:Y? . . . . . SPA 5-28

:CALCulate:MARKer{1|2|3|4|5|6}:Y? . . . . . AM/FM/PM 6-3

:CALCulate:MARKer{1|2|3|4|5|6}[:SET]:CENTer . . . . . SPA 5-29

:CALCulate:MARKer{1|2|3|4|5|6}[:SET]:CENTer . . . . . AM/FM/PM 6-2

:CALCulate:MARKer{1|2|3|4|5|6}[:SET]:RLEVel . . . . . SPA 5-29
:CALCulate:MARKer{1|2|3|4|5|6}[:SET]:RLEVel . . . . . AM/FM/PM 6-3
:CALCulate:MARKer{1|2|3|4|5|6}[:STATe] OFF|ON|0|1
:CALCulate:MARKer{1|2|3|4|5|6}[:STATe]?. . . . . AM/FM/PM 6-1
:CALCulate:MARKer{1|2|3|4|5|6}[:STATe] . . . . . SPA 5-29
:CALCulate:MARKer<Mk>:DOMain? . . . . . VNA 3-33
:CALCulate:MARKer<Mk>:FORMat . . . . . VNA 3-34
:CALCulate:MARKer<Mk>:REFerence[:STATe] . . . . . VNA 3-36
:CALCulate:MARKer<Mk>:SOURce . . . . . VNA 3-37
:CALCulate:MATH:FUNCtion . . . . . VNA 3-42
:CALCulate:MATH:MEMorize . . . . . VNA 3-42
:CALCulate:TRANsform:DISTance:MAXimum?. . . . . VNA 3-44
:CALCulate:TRANsform:DISTance:RESolution? . . . . . VNA 3-44
:CALCulate:TRANsform:DISTance:STARt . . . . . VNA 3-45
:CALCulate:TRANsform:DISTance:STOP. . . . . VNA 3-45
:CALCulate:TRANsform:DISTance:UNIT . . . . . VNA 3-46
:CALCulate<Tr>:DATA?. . . . . VNA 3-3
:CALCulate<Tr>:FORMat . . . . . VNA 3-5
:CALCulate<Tr>:LIMit:FAIL? . . . . . VNA 5-3
:CALCulate<Tr>:LIMit:LOWer:FAIL?. . . . . VNA 5-4
:CALCulate<Tr>:LIMit:LOWer:POINt:ADD. . . . . VNA 3-7
:CALCulate<Tr>:LIMit:LOWer:POINt:DELete . . . . . VNA 3-7
:CALCulate<Tr>:LIMit:LOWer:POINt:LEFT . . . . . VNA 3-8
:CALCulate<Tr>:LIMit:LOWer:POINt:RIGHt . . . . . VNA 3-8
:CALCulate<Tr>:LIMit:LOWer:POINt:X . . . . . VNA 3-9
:CALCulate<Tr>:LIMit:LOWer:POINt:Y . . . . . VNA 3-10
:CALCulate<Tr>:LIMit:LOWer:POINt? . . . . . VNA 3-6
:CALCulate<Tr>:LIMit:LOWer:X . . . . . VNA 3-12
:CALCulate<Tr>:LIMit:LOWer:Y . . . . . VNA 3-13
:CALCulate<Tr>:LIMit:LOWer[:STATe] . . . . . VNA 3-11
:CALCulate<Tr>:LIMit:PFMessage . . . . . VNA 5-5

:CALCulate<Tr>:LIMit:UPPer:FAIL? . . . . . VNA 5-6
:CALCulate<Tr>:LIMit:UPPer:POINt:ADD . . . . . VNA 3-20
:CALCulate<Tr>:LIMit:UPPer:POINt:DELete. . . . . VNA 3-21
:CALCulate<Tr>:LIMit:UPPer:POINt:LEFT . . . . . VNA 3-21
:CALCulate<Tr>:LIMit:UPPer:POINt:RIGHt . . . . . VNA 3-22
:CALCulate<Tr>:LIMit:UPPer:POINt:X. . . . . VNA 3-23
:CALCulate<Tr>:LIMit:UPPer:POINt:Y. . . . . VNA 3-24
:CALCulate<Tr>:LIMit:UPPer:POINt?. . . . . VNA 3-20
:CALCulate<Tr>:LIMit:UPPer:X. . . . . VNA 3-26
:CALCulate<Tr>:LIMit:UPPer:Y. . . . . VNA 3-27
:CALCulate<Tr>:LIMit:UPPer[:STATe] . . . . . VNA 3-25
:CALCulate<Tr>:SMOothing:APERture . . . . . VNA 3-43
:CALCulate<Tr>:TRANsform:DISTance:DATA? . . . . . VNA 3-46
:CONFigure:ACPower. . . . . SPA 5-30
:CONFigure:CHPower . . . . . SPA 5-30
:CONFigure:FSTRength. . . . . SPA 5-31
:CONFigure:OBWidth . . . . . SPA 5-31
:CONFigure? . . . . . SPA 5-32
:DISPlay:GRID . . . . . SPA 5-33
:DISPlay:WINDow:TRACe:Y:ADJust . . . . . AM/FM/PM 6-5
:DISPlay:WINDow:TRACe:Y:ADJust . . . . . SPA 5-33
:DISPlay:WINDow:TRACe:Y:AFPanalyzer:PWR:OFFSet <rel ampl>
:DISPlay:WINDow:TRACe:Y:AFPanalyzer:PWR:OFFSet? . . AM/FM/PM 6-5
:DISPlay:WINDow:TRACe:Y[:SCALe]:GDAPerture. . . . . VNA 3-48
:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision <rel ampl>
:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision?. . . . . AM/FM/PM 6-5
:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision . . . . . SPA 5-33
:DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel . . . . . SPA 5-34
:DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel:OFFSet . . . . . SPA 5-34
:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:PDIVision . . . . . VNA 3-49
:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:RLEVel . . . . . VNA 3-50

:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:RPOSition . . . . . VNA 3-51
:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:SMCHart . . . . . VNA 3-52
:DISPlay[:WINDow]:TRACe TRACe|MEMory|BOTH . . . . . VNA 3-47
:DISPlay[:WINDow]:TRACe:FORMat . . . . . VNA 3-47
:FETCh:ACPower?. . . . . SPA 5-35
:FETCh:CHPower:CHPower? . . . . . SPA 5-35
:FETCh:CHPower:DENSity? . . . . . SPA 5-36
:FETCh:CHPower? . . . . . SPA 5-36
:FETCh:GPS?. . . . . All Modes 7-14
:FETCh:OBWidth:FREQuency? . . . . . SPA 5-37
:FETCh:OBWidth? . . . . . SPA 5-37
:FETCh:VVM:DATA? . . . . . VVM 4-18
:FETCh:VVM:REFerence:DATA?. . . . . VVM 4-20
:FORMat[:READings][:DATA] ASCii|INTeger,32|REAL,[<length>]
:FORMat[:READings][:DATA]? . . . . . AM/FM/PM 6-6
:FORMat[:READings][:DATA] . . . . . SPA 5-38
:FORMat[:READings][:DATA] . . . . . VNA 3-53
:INITiate:CONTinuous OFF|ON|0|1
:INITiate:CONTinuous? . . . . . AM/FM/PM 6-8
:INITiate:CONTinuous . . . . . SPA 5-40
:INITiate:CONTinuous . . . . . VNA 3-54
:INITiate:HOLD . . . . . VNA 3-55
:INITiate:SAVe:ON:EVENt:CROSsing:LIMit OFF|ON|0|1 . . . . . SPA 5-41
:INITiate:SAVe:ON:EVENt:SWEep OFF|ON|0|1 . . . . . SPA 5-41
:INITiate:SAVe:THEn:STOp OFF|ON|0|1 . . . . . SPA 5-41
:INITiate[:IMMediate]. . . . . SPA 5-40
:INITiate[:IMMediate]. . . . . VNA 3-55
:INITiate[:IMMediate]. . . . . AM/FM/PM 6-8
:INPut:BIAS:INTernal:TRIPped[:STATe]? . . . . . VNA 3-56
:INPut:BIAS:STATe . . . . . VNA 3-57
:INPut<port_no>:BIAS:CURRent. . . . . VNA 3-56

:INPut<port_no>:BIAS:VOLTage . . . . . VNA 3-57
:INSTrument:CATalog:FULL? . . . . . All Modes 7-2
:INSTrument:NSELect . . . . . All Modes 7-3
:INSTrument[:SELect] . . . . . All Modes 7-4
:MEASure:ACPower? . . . . . SPA 5-42
:MEASure:CHPower:CHPower? . . . . . SPA 5-43
:MEASure:CHPower:DENSity? . . . . . SPA 5-44
:MEASure:CHPower? . . . . . SPA 5-44
:MEASure:OBWidth? . . . . . SPA 5-45
:MMEMory:DATA? . . . . . All Modes 7-6
:MMEMory:DATA? . . . . . SPA 5-46
:MMEMory:DELete <filename> . . . . . AM/FM/PM 6-9
:MMEMory:DELete . . . . . All Modes 7-6
:MMEMory:DELete . . . . . SPA 5-46
:MMEMory:LOAD:LIMit . . . . . SPA 5-47
:MMEMory:LOAD:STATe <integer>,<filename> . . . . . AM/FM/PM 6-9
:MMEMory:LOAD:STATe . . . . . SPA 5-47
:MMEMory:LOAD:STATe . . . . . VNA 3-59
:MMEMory:LOAD:STATe . . . . . VVM 4-4
:MMEMory:LOAD:TRACe <integer>,<filename> . . . . . AM/FM/PM 6-9
:MMEMory:LOAD:TRACe . . . . . SPA 5-48
:MMEMory:LOAD:TRACe . . . . . VNA 3-60
:MMEMory:LOAD:TRACe . . . . . VVM 4-5
:MMEMory:MSIS . . . . . All Modes 7-8
:MMEMory:MSIS:COPY . . . . . All Modes 7-8
:MMEMory:MSIS:DESTination . . . . . All Modes 7-9
:MMEMory:STORe:JPEG . . . . . All Modes 7-10
:MMEMory:STORe:LIMit . . . . . SPA 5-49
:MMEMory:STORe:STATe <integer>,<filename> . . . . . AM/FM/PM 6-10
:MMEMory:STORe:STATe . . . . . SPA 5-49

:MMEMory:STORe:STATe . . . . . VNA 3-61
:MMEMory:STORe:STATe . . . . . VVM 4-6
:MMEMory:STORe:TRACe <integer>,<filename> . . . . . AM/FM/PM 6-11
:MMEMory:STORe:TRACe . . . . . SPA 5-50
:MMEMory:STORe:TRACe . . . . . VNA 3-62
:MMEMory:STORe:TRACe . . . . . VVM 4-7
:READ:ACPower? . . . . . SPA 5-51
:READ:CHPower:DENSity? . . . . . SPA 5-52
:READ:CHPower? . . . . . SPA 5-52
:READ:OBWidth? . . . . . SPA 5-53
:SENSe<Tr>:FREQuency:DATA? . . . . . VNA 3-94
:SOURce:CORRection:RVELocity . . . . . VNA 3-101
:SOURce:CORRection:RVELocity:CABLoss . . . . . VNA 3-101
:SOURce:POWer . . . . . VNA 3-100
:STATus:OPERation? . . . . . VNA 3-102
:SYSTem:ALTernate:STATe . . . . . VNA 3-103
:SYSTem:MBTemperature? . . . . . VNA 3-103
:SYSTem:OPTions? . . . . . All Modes 7-11
:SYSTem:PRESet . . . . . All Modes 7-11
:SYSTem:PRESet . . . . . VNA 3-104
:SYSTem:WINDow:CLOSe . . . . . All Modes 7-14
:TRACe:COPY . . . . . SPA 5-54
:TRACe:EXCHange TRACE2,TRACE3 . . . . . SPA 5-54
:TRACe:PREamble? . . . . . SPA 5-55
:TRACe:PREamble? . . . . . VNA 3-106
:TRACe:PREamble? . . . . . VVM 4-8
:TRACe[:DATA] . . . . . SPA 5-62
:TRACe[:DATA]? . . . . . VNA 3-105
:TRACe[:DATA]? . . . . . AM/FM/PM 6-12
:TRACe{1|2|3}:DISPlay[:STATe] . . . . . SPA 5-63

:TRACe{1|2|3}:WRITe[:STATe] . . . . . SPA 5-64
:TRACe1:OPERation. . . . . SPA 5-65
:TRACe2:OPERation. . . . . SPA 5-66
:TRACe3:OPERation. . . . . SPA 5-67
:TRIGger[:SEQuence]:SOURce . . . . . SPA 5-68
:TRIGger[:SEQuence]:VIDeo:LEVel . . . . . SPA 5-68
:TRIGger[:SEQuence]:VIDeo:POSition . . . . . SPA 5-69
:UNIT:POWer . . . . . SPA 5-70
[:SENSe]:ACPower:BANDwidth|BWIDth:ADJacent. . . . . SPA 5-71
[:SENSe]:ACPower:BANDwidth|BWIDth:MAIN. . . . . SPA 5-71
[:SENSe]:ACPower:BANDwidth|BWIDth:SPACing . . . . . SPA 5-72
[:SENSe]:ACPower:STATe . . . . . SPA 5-72
[:SENSe]:AFPanalyzer:AVERage:COUNt <avg count>
[:SENSe]:AFPanalyzer:AVERage:COUNt? . . . . . AM/FM/PM 6-13
[:SENSe]:AFPanalyzer:DEMod:DATA?. . . . . AM/FM/PM 6-13
[:SENSe]:AFPanalyzer:DEMod:MODE RFSP|AFSP|AFWV|SUMMary
[:SENSe]:AFPanalyzer:DEMod:MODE? . . . . . AM/FM/PM 6-13
[:SENSe]:AFPanalyzer:DEMod:TYPE AM|FM|PM
[:SENSe]:AFPanalyzer:DEMod:TYPE? . . . . . AM/FM/PM 6-14
[:SENSe]:AFPanalyzer:FM:SCALe <percentage>
[:SENSe]:AFPanalyzer:FM:SCALe? . . . . . AM/FM/PM 6-14
[:SENSe]:AFPanalyzer:IFBW <freq>
[:SENSe]:AFPanalyzer:IFBW?. . . . . AM/FM/PM 6-14
[:SENSe]:AFPanalyzer:IFBW:AUTO 0|1
[:SENSe]:AFPanalyzer:IFBW:AUTO? . . . . . AM/FM/PM 6-15
[:SENSe]:AFPanalyzer:PM:SCALe <Radians>
[:SENSe]:AFPanalyzer:PM:SCALe? . . . . . AM/FM/PM 6-15
[:SENSe]:AFPanalyzer:RFSPectrum:DATA? . . . . . AM/FM/PM 6-15
[:SENSe]:AFPanalyzer:SPAN <freq>
[:SENSe]:AFPanalyzer:SPAN?. . . . . AM/FM/PM 6-16
[:SENSe]:AFPanalyzer:SUMMary:DATA? . . . . . AM/FM/PM 6-16
[:SENSe]:AFPanalyzer:SWEep:TIME <time>
[:SENSe]:AFPanalyzer:SWEep:TIME? . . . . . AM/FM/PM 6-16

[:SENSe]:APPLication:TST:RESult? . . . . . VNA 3-64
[:SENSe]:APPLication:TST? . . . . . VNA 3-64
[:SENSe]:AVERage:CLEar . . . . . VNA 3-65
[:SENSe]:AVERage:COUNt . . . . . SPA 5-72
[:SENSe]:AVERage:COUNt . . . . . VNA 3-65
[:SENSe]:AVERage:TYPE. . . . . SPA 5-73
[:SENSe]:BANDwidth|BWIDth:VIDeo . . . . . SPA 5-75
[:SENSe]:BANDwidth|BWIDth:VIDeo:AUTO . . . . . SPA 5-76
[:SENSe]:BANDwidth|BWIDth:VIDeo:RATio . . . . . SPA 5-76
[:SENSe]:BANDwidth|BWIDth[:RESolution]. . . . . SPA 5-74
[:SENSe]:BANDwidth|BWIDth[:RESolution]:AUTO. . . . . SPA 5-74
[:SENSe]:BANDwidth|BWIDth[:RESolution]:RATio. . . . . SPA 5-75
[:SENSe]:CALibration:STATe?. . . . . VNA 3-66
[:SENSe]:CHPower:BANDwidth|BWIDth:INTegration . . . . . SPA 5-77
[:SENSe]:CHPower:STATe . . . . . SPA 5-77
[:SENSe]:CORRection:CKIT:INFormation? . . . . . VNA 3-70
[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX:SOLT:C0|1|2|3 VNA 3-75
[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX:SOLT:L0|1|2|3 VNA 3-75
[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX:SOLT:NAME VNA 3-76
[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX:SOLT:OPEN VNA 3-76
[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX:SOLT:SHORt VNA 3-77
[:SENSe]:CORRection:COLLect:ABORt:ALL . . . . . VNA 3-78
[:SENSe]:CORRection:COLLect:ACQUire:STATus?. . . . . VNA 3-81
[:SENSe]:CORRection:COLLect:CONNector<port_no> . . . . . VNA 3-81
[:SENSe]:CORRection:COLLect:INTerpolation[:STATe] . . . . . VNA 3-85
[:SENSe]:CORRection:COLLect:MEDium. . . . . VNA 3-85
[:SENSe]:CORRection:COLLect:METHod. . . . . VNA 3-86
[:SENSe]:CORRection:COLLect:SAVe. . . . . VNA 3-86
[:SENSe]:CORRection:COLLect:STATus:ACCuracy?. . . . . VNA 3-87

[:SENSe]:CORRection:COLLect:STATus? . . . . . VNA 3-86
[:SENSe]:CORRection:COLLect:TYPE . . . . . VNA 3-88
[:SENSe]:CORRection:COLLect[:ACQUire] . . . . . VNA 3-79
[:SENSe]:CORRection:DATA? . . . . . VNA 3-68
[:SENSe]:CORRection:IMPedance[:INPut]:OFFSet . . . . . SPA 5-78
[:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude] . . . . . SPA 5-78
[:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude]:SMCHart VNA 3-69
[:SENSe]:CORRection[:STATe] . . . . . VNA 3-69
[:SENSe]:DETector[:FUNCtion] . . . . . SPA 5-79
[:SENSe]:DITHering:[STATe]OFF|ON|0|1 . . . . . VNA 3-89
[:SENSe]:EXTRefstatus? . . . . . SPA 5-79
[:SENSe]:FREQuency:CENTer <freq>
[:SENSe]:FREQuency:CENTer? . . . . . AM/FM/PM 6-17
[:SENSe]:FREQuency:CENTer . . . . . SPA 5-80
[:SENSe]:FREQuency:CENTer . . . . . VNA 3-90
[:SENSe]:FREQuency:DSPAn? . . . . . VNA 3-91
[:SENSe]:FREQuency:SIGStandard:CHANnel <number>
[:SENSe]:FREQuency:SIGStandard:CHANnel? . . . . . AM/FM/PM 6-17
[:SENSe]:FREQuency:SIGStandard:CHANnel . . . . . SPA 5-80
[:SENSe]:FREQuency:SIGStandard:NAMe <string>
[:SENSe]:FREQuency:SIGStandard:NAMe? . . . . . AM/FM/PM 6-17
[:SENSe]:FREQuency:SIGStandard:NAMe . . . . . SPA 5-81
[:SENSe]:FREQuency:SPAN <freq>
[:SENSe]:FREQuency:SPAN? . . . . . AM/FM/PM 6-18
[:SENSe]:FREQuency:SPAN . . . . . SPA 5-81
[:SENSe]:FREQuency:SPAN . . . . . VNA 3-91
[:SENSe]:FREQuency:SPAN:FULL . . . . . SPA 5-82
[:SENSe]:FREQuency:SPAN:FULL . . . . . AM/FM/PM 6-18
[:SENSe]:FREQuency:SPAN:MINimum . . . . . AM/FM/PM 6-18
[:SENSe]:FREQuency:SPAN:PREVious . . . . . SPA 5-82
[:SENSe]:FREQuency:SPAN:PREVious . . . . . AM/FM/PM 6-18
[:SENSe]:FREQuency:STARt . . . . . SPA 5-82

[:SENSe]:FREQuency:STARt . . . . . VNA 3-92

[:SENSe]:FREQuency:STEP[:INCRement] <freq>
[:SENSe]:FREQuency:STEP[:INCRement]? . . . . . AM/FM/PM 6-18

[:SENSe]:FREQuency:STEP[:INCRement] . . . . . SPA 5-83

[:SENSe]:FREQuency:STOP. . . . . SPA 5-83

[:SENSe]:FREQuency:STOP. . . . . VNA 3-93

[:SENSe]:FSTRength:ANTenna . . . . . SPA 5-83

[:SENSe]:FSTRength:STATe . . . . . SPA 5-84

[:SENSe]:GPS . . . . . All Modes 7-12

[:SENSe]:GPS:CURRent? . . . . . All Modes 7-12

[:SENSe]:GPS:RESet. . . . . All Modes 7-12

[:SENSe]:GPS:VOLTage . . . . . All Modes 7-13

[:SENSe]:OBWidth:METHod XDB | PERCent
[:SENSe]:OBWidth:METHod? . . . . . AM/FM/PM 6-19

[:SENSe]:OBWidth:METHod . . . . . SPA 5-85

[:SENSe]:OBWidth:PERCent <percentage>
[:SENSe]:OBWidth:PERCent? . . . . . AM/FM/PM 6-19

[:SENSe]:OBWidth:PERCent . . . . . SPA 5-85

[:SENSe]:OBWidth:STATe . . . . . SPA 5-86

[:SENSe]:OBWidth:XDB <rel ampl>
[:SENSe]:OBWidth:XDB? . . . . . AM/FM/PM 6-20

[:SENSe]:OBWidth:XDB. . . . . SPA 5-86

[:SENSe]:POWer[:RF]:ATTenuation . . . . . SPA 5-87

[:SENSe]:POWer[:RF]:ATTenuation:AUTO . . . . . SPA 5-87

[:SENSe]:POWer[:RF]:GAIN[:STATe] . . . . . SPA 5-88

[:SENSe]:RFON[:STATe] . . . . . VNA 3-63

[:SENSe]:SWEep:IFBW . . . . . VNA 3-95

[:SENSe]:SWEep:IFBW . . . . . VNA 4-11

[:SENSe]:SWEep:POINts . . . . . VNA 3-96

[:SENSe]:SWEep:STATus? . . . . . SPA 5-88

[:SENSe]:SWEep:TIME. . . . . SPA 5-88

[:SENSe]:SWEep:TRACe?. . . . . SPA 5-89

[:SENSe]:SWEep:TYPE . . . . . VNA 3-96
[:SENSe]:TRACe:SELect?. . . . . VNA 3-99
[:SENSe]:TRACe:TOTal . . . . . VNA 3-98
[:SENSe]:TRACe<Tr>:DOMain . . . . . VNA 3-97
[:SENSe]:TRACe<Tr>:SELect . . . . . VNA 3-97
[:SENSe]:TRACe<Tr>:SPARams . . . . . VNA 3-98
[:SENSe]:VVM:CABLe:SELect. . . . . VVM 4-14
[:SENSe]:VVM:FORMat . . . . . VVM 4-12
[:SENSe]:VVM:FREQuency:CW . . . . . VVM 4-15
[:SENSe]:VVM:MODE CW|TABLe . . . . . VVM 4-13
[:SENSe]:VVM:REFerence:CLEar . . . . . VVM 4-16
[:SENSe]:VVM:REFerence:MEMorize. . . . . VVM 4-16
[:SENSe]:VVM:TYPE . . . . . VVM 4-13
MMEM:DIR? . . . . . All Modes 7-7

Appendix C — SCPI Commands by Mode

General Information 1

Programming with SCPI 1

VNA Commands 1

:CALCulate<Tr>:DATA? . . . . . VNA 3-3
:CALCulate<Tr>:FORMat . . . . . VNA 3-5
:CALCulate:LIMit:ALARm. . . . . VNA 3-6
:CALCulate<Tr>:LIMit:LOWer:POINt? . . . . . VNA 3-6
:CALCulate<Tr>:LIMit:LOWer:POINt:ADD . . . . . VNA 3-7
:CALCulate<Tr>:LIMit:LOWer:POINt:DELete . . . . . VNA 3-7
:CALCulate<Tr>:LIMit:LOWer:POINt:LEFT. . . . . VNA 3-8
:CALCulate<Tr>:LIMit:LOWer:POINt:RIGHt . . . . . VNA 3-8
:CALCulate<Tr>:LIMit:LOWer:POINt:X . . . . . VNA 3-9
:CALCulate<Tr>:LIMit:LOWer:POINt:Y . . . . . VNA 3-10
:CALCulate<Tr>:LIMit:LOWer[:STATe]. . . . . VNA 3-11
:CALCulate<Tr>:LIMit:LOWer:X . . . . . VNA 3-12
:CALCulate<Tr>:LIMit:LOWer:Y . . . . . VNA 3-13
:CALCulate:LIMit:POINt? . . . . . VNA 3-13
:CALCulate:LIMit:POINt:ADD . . . . . VNA 3-14
:CALCulate:LIMit:POINt:DELete . . . . . VNA 3-14
:CALCulate:LIMit:POINt:LEFT . . . . . VNA 3-15
:CALCulate:LIMit:POINt:RIGHt. . . . . VNA 3-15
:CALCulate:LIMit:POINt:X . . . . . VNA 3-16
:CALCulate:LIMit:POINt:Y . . . . . VNA 3-17
:CALCulate:LIMit[:STATe] . . . . . VNA 3-18
:CALCulate:LIMit:TYPE . . . . . VNA 3-19
:CALCulate<Tr>:LIMit:UPPer:POINt?. . . . . VNA 3-20

:CALCulate<Tr>:LIMit:UPPer:POINt:ADD . . . . . VNA 3-20
:CALCulate<Tr>:LIMit:UPPer:POINt:DELete . . . . . VNA 3-21
:CALCulate<Tr>:LIMit:UPPer:POINt:LEFT . . . . . VNA 3-21
:CALCulate<Tr>:LIMit:UPPer:POINt:RIGHt. . . . . VNA 3-22
:CALCulate<Tr>:LIMit:UPPer:POINt:X . . . . . VNA 3-23
:CALCulate<Tr>:LIMit:UPPer:POINt:Y . . . . . VNA 3-24
:CALCulate<Tr>:LIMit:UPPer[:STATe] . . . . . VNA 3-25
:CALCulate<Tr>:LIMit:UPPer:X . . . . . VNA 3-26
:CALCulate<Tr>:LIMit:UPPer:Y . . . . . VNA 3-27
:CALCulate:LIMit:X . . . . . VNA 3-28
:CALCulate:LIMit:Y . . . . . VNA 3-29
:CALCulate:MARKer:AOFF . . . . . VNA 3-30
:CALCulate:MARKer:DATA? . . . . . VNA 3-30
:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:DELTa:REFerence . VNA 3-31
:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:DELTa[:STATe] VNA 3-32
:CALCulate:MARKer:DISPlay:FORMat . . . . . VNA 3-33
:CALCulate:MARKer<Mk>:DOMain? . . . . . VNA 3-33
:CALCulate:MARKer<Mk>:FORMat . . . . . VNA 3-34
:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:MAXimum VNA 3-35
:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:MINimum VNA 3-35
:CALCulate:MARKer<Mk>:REFerence[:STATe] . . . . . VNA 3-36
:CALCulate:MARKer<Mk>:SOURce . . . . . VNA 3-37
:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:TYPE . . . VNA 3-38
:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:X. . . . . VNA 3-39
:CALCulate:MARKer[1]|2|3|4|5|6|7|8|9|10|11|12:Y?. . . . . VNA 3-40
:CALCulate:MATH:FUNCtion . . . . . VNA 3-42
:CALCulate:MATH:MEMorize . . . . . VNA 3-42
:CALCulate<Tr>:SMOothing:APERture . . . . . VNA 3-43
:CALCulate:TRANsform:DISTance:MAXimum?. . . . . VNA 3-44

:CALCulate:TRANsform:DISTance:RESolution? . . . . . VNA 3-44
:CALCulate:TRANsform:DISTance:STARt . . . . . VNA 3-45
:CALCulate:TRANsform:DISTance:STOP . . . . . VNA 3-45
:CALCulate:TRANsform:DISTance:UNIT. . . . . VNA 3-46
:CALCulate<Tr>:TRANsform:DISTance:DATA? . . . . . VNA 3-46
:DISPlay[:WINDow]:TRACe TRACe | MEMory | BOTH . . . . . VNA 3-47
:DISPlay[:WINDow]:TRACe:FORMat . . . . . VNA 3-47
:DISPlay:WINDow:TRACe:Y[:SCALe]:GDAPerture. . . . . VNA 3-48
:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:PDIVision . . . . . VNA 3-49
:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:RLEVel . . . . . VNA 3-50
:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:RPOSition . . . . . VNA 3-51
:DISPlay:WINDow:TRACe<Tr>:Y[:SCALe]:SMCHart . . . . . VNA 3-52
:FORMat[:READings][:DATA] . . . . . VNA 3-53
:INITiate:CONTinuous . . . . . VNA 3-54
:INITiate:HOLD . . . . . VNA 3-55
:INITiate[:IMMediate] . . . . . VNA 3-55
:INPut<port_no>:BIAS:CURRent . . . . . VNA 3-56
:INPut:BIAS:INTernal:TRIPped[:STATe]? . . . . . VNA 3-56
:INPut:BIAS:STATe . . . . . VNA 3-57
:INPut<port_no>:BIAS:VOLTage . . . . . VNA 3-57
:MMEMory:LOAD:STATe . . . . . VNA 3-59
:MMEMory:LOAD:TRACe . . . . . VNA 3-60
:MMEMory:STORe:STATe . . . . . VNA 3-61
:MMEMory:STORe:TRACe. . . . . VNA 3-62
[:SENSe]:RFON[:STATe] . . . . . VNA 3-63
[:SENSe]:APPLication:TST?. . . . . VNA 3-64
[:SENSe]:APPLication:TST:RESult? . . . . . VNA 3-64
[:SENSe]:AVERage:CLEar . . . . . VNA 3-65
[:SENSe]:AVERage:COUNt . . . . . VNA 3-65
[:SENSe]:CALibration:STATe? . . . . . VNA 3-66

[:SENSe]:CORRection:DATA? . . . . . VNA 3-68
[:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude]:SMCHart VNA 3-69
[:SENSe]:CORRection[:STATe] . . . . . VNA 3-69
[:SENSe]:CORRection:CKIT:INFormation? . . . . . VNA 3-70
[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX:SOLT:L0|1|2|3 VNA 3-75
[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX:SOLT:C0|1|2|3 VNA 3-75
[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX:SOLT:NAME VNA 3-76
[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX:SOLT:OPEN VNA 3-76
[:SENSe]:CORRection:CKIT:USER[1]|2|3|4:COAX:SOLT:SHORt VNA 3-77
[:SENSe]:CORRection:COLLect:ABORt:ALL . . . . . VNA 3-78
[:SENSe]:CORRection:COLLect[:ACQUire] . . . . . VNA 3-79
[:SENSe]:CORRection:COLLect:ACQUire:STATus?. . . . . VNA 3-81
[:SENSe]:CORRection:COLLect:CONNector<port_no> . . . . . VNA 3-81
[:SENSe]:CORRection:COLLect:INTerpolation[:STATe] . . . . . VNA 3-85
[:SENSe]:CORRection:COLLect:MEDium . . . . . VNA 3-85
[:SENSe]:CORRection:COLLect:METHod . . . . . VNA 3-86
[:SENSe]:CORRection:COLLect:SAVe . . . . . VNA 3-86
[:SENSe]:CORRection:COLLect:STATus? . . . . . VNA 3-86
[:SENSe]:CORRection:COLLect:STATus:ACCuracy? . . . . . VNA 3-87
[:SENSe]:CORRection:COLLect:TYPE . . . . . VNA 3-88
[:SENSe]:DITHering:[STATe]OFF|ON|0|1. . . . . VNA 3-89
[:SENSe]:FREQuency:CENTer. . . . . VNA 3-90
[:SENSe]:FREQuency:SPAN . . . . . VNA 3-91
[:SENSe]:FREQuency:DSPAn?. . . . . VNA 3-91
[:SENSe]:FREQuency:STARt . . . . . VNA 3-92
[:SENSe]:FREQuency:STOP. . . . . VNA 3-93
:SENSe<Tr>:FREQuency:DATA?. . . . . VNA 3-94
[:SENSe]:SWEep:IFBW . . . . . VNA 3-95
[:SENSe]:SWEep:POINts . . . . . VNA 3-96

[:SENSe]:SWEep:TYPE . . . . . VNA 3-96
[:SENSe]:TRACe<Tr>:DOMain . . . . . VNA 3-97
[:SENSe]:TRACe<Tr>:SELect . . . . . VNA 3-97
[:SENSe]:TRACe<Tr>:SPARams . . . . . VNA 3-98
[:SENSe]:TRACe:TOTal . . . . . VNA 3-98
[:SENSe]:TRACe:SELect?. . . . . VNA 3-99
:SOURce:POWer . . . . . VNA 3-100
:SOURce:CORRection:RVELocity . . . . . VNA 3-101
:SOURce:CORRection:RVELocity:CABLoss . . . . . VNA 3-101
:STATus:OPERation? . . . . . VNA 3-102
:SYSTem:ALTernate:STATe . . . . . VNA 3-103
:SYSTem:MBTemperature? . . . . . VNA 3-103
:SYSTem:PRESet . . . . . VNA 3-104
:TRACe[:DATA]?. . . . . VNA 3-105
:TRACe:PREamble? . . . . . VNA 3-106

Vector Voltmeter Commands 1

:MMEMory:LOAD:STATe . . . . . VVM 4-4
:MMEMory:LOAD:TRACe . . . . . VVM 4-5
:MMEMory:STORe:STATe . . . . . VVM 4-6
:MMEMory:STORe:TRACe. . . . . VVM 4-7
:TRACe:PREamble? . . . . . VVM 4-8
[:SENSe]:SWEep:IFBW . . . . . VNA 4-11
[:SENSe]:VVM:FORMat . . . . . VVM 4-12
[:SENSe]:VVM:MODE CW|TABLe . . . . . VVM 4-13
[:SENSe]:VVM:TYPE . . . . . VVM 4-13
[:SENSe]:VVM:CABLe:SELect. . . . . VVM 4-14
[:SENSe]:VVM:FREQuency:CW . . . . . VVM 4-15
[:SENSe]:VVM:REFerence:CLEar . . . . . VVM 4-16
[:SENSe]:VVM:REFerence:MEMorize . . . . . VVM 4-16
:FETCh:VVM:DATA? . . . . . VVM 4-18

:FETCh:VVM:REFerence:DATA? . . . . . VVM 4-20

Spectrum Analyzer Commands 1

:ABORt . . . . . SPA 5-2
:CALCulate:LIMit:ALARm . . . . . SPA 5-3
:CALCulate<Tr>:LIMit:FAIL? . . . . . VNA 5-3
:CALCulate<Tr>:LIMit:LOWer:FAIL? . . . . . VNA 5-4
:CALCulate<Tr>:LIMit:PFMessage . . . . . VNA 5-5
:CALCulate<Tr>:LIMit:UPPer:FAIL? . . . . . VNA 5-6
:CALCulate:LIMit:CENTer . . . . . SPA 5-6
:CALCulate:LIMit:ENVelope:CREate . . . . . SPA 5-7
:CALCulate:LIMit:ENVelope:OFFSet . . . . . SPA 5-7
:CALCulate:LIMit:ENVelope:POINt . . . . . SPA 5-8
:CALCulate:LIMit:ENVelope:SHAPe . . . . . SPA 5-8
:CALCulate:LIMit:ENVelope:UPDate:X . . . . . SPA 5-9
:CALCulate:LIMit:ENVelope:UPDate:Y . . . . . SPA 5-9
:CALCulate:LIMit:FAIL? . . . . . SPA 5-9
:CALCulate:LIMit:LOWer:ALARm . . . . . SPA 5-10
:CALCulate:LIMit:LOWer:FAIL? . . . . . SPA 5-10
:CALCulate:LIMit:LOWer:POINt? . . . . . SPA 5-10
:CALCulate:LIMit:LOWer[:STATe] . . . . . SPA 5-11
:CALCulate:LIMit:LTYPe . . . . . SPA 5-12
:CALCulate:LIMit:MIRRor . . . . . SPA 5-12
:CALCulate:LIMit:POINt:ADD . . . . . SPA 5-12
:CALCulate:LIMit:POINt:DELete . . . . . SPA 5-13
:CALCulate:LIMit:POINt:LEFT . . . . . SPA 5-13
:CALCulate:LIMit:POINt:RIGHt . . . . . SPA 5-13
:CALCulate:LIMit:POINt:X . . . . . SPA 5-14
:CALCulate:LIMit:POINt:Y . . . . . SPA 5-14
:CALCulate:LIMit:POINt? . . . . . SPA 5-15
:CALCulate:LIMit:TYPe . . . . . SPA 5-15

:CALCulate:LIMit:UPPer:ALARm . . . . . SPA 5-16
:CALCulate:LIMit:UPPer:FAIL? . . . . . SPA 5-16
:CALCulate:LIMit:UPPer:POINt? . . . . . SPA 5-16
:CALCulate:LIMit:UPPer[:STATe] . . . . . SPA 5-17
:CALCulate:LIMit:VALue . . . . . SPA 5-17
:CALCulate:LIMit:VERTical:ADD. . . . . SPA 5-18
:CALCulate:LIMit:X . . . . . SPA 5-18
:CALCulate:LIMit:Y . . . . . SPA 5-19
:CALCulate:LIMit[:SET]:DEFault. . . . . SPA 5-20
:CALCulate:LIMit[:STATe] . . . . . SPA 5-20
:CALCulate:MARKer:AOFF. . . . . SPA 5-21
:CALCulate:MARKer:PEAK:THReshold. . . . . SPA 5-21
:CALCulate:MARKer:TABLe:DATA? . . . . . SPA 5-21
:CALCulate:MARKer:TABLe[:STATe] . . . . . SPA 5-22
:CALCulate:MARKer{1|2|3|4|5|6}:DELTa:NOISe[:STATe] . . . . SPA 5-22
:CALCulate:MARKer{1|2|3|4|5|6}:DELTa:X . . . . . SPA 5-23
:CALCulate:MARKer{1|2|3|4|5|6}:DELTa:Y? . . . . . SPA 5-23
:CALCulate:MARKer{1|2|3|4|5|6}:DELTa[:SET]:SPAN . . . . . SPA 5-24
:CALCulate:MARKer{1|2|3|4|5|6}:DELTa[:STATe]. . . . . SPA 5-24
:CALCulate:MARKer{1|2|3|4|5|6}:FCOunt[:STATe] . . . . . SPA 5-25
:CALCulate:MARKer{1|2|3|4|5|6}:FIXed[:STATe]. . . . . SPA 5-25
:CALCulate:MARKer{1|2|3|4|5|6}:MAXimum . . . . . SPA 5-26
:CALCulate:MARKer{1|2|3|4|5|6}:MAXimum:LEFT. . . . . SPA 5-26
:CALCulate:MARKer{1|2|3|4|5|6}:MAXimum:NEXT . . . . . SPA 5-26
:CALCulate:MARKer{1|2|3|4|5|6}:MAXimum:RIGHt . . . . . SPA 5-27
:CALCulate:MARKer{1|2|3|4|5|6}:NOISe[:STATe] . . . . . SPA 5-27
:CALCulate:MARKer{1|2|3|4|5|6}:X. . . . . SPA 5-28
:CALCulate:MARKer{1|2|3|4|5|6}:Y? . . . . . SPA 5-28
:CALCulate:MARKer{1|2|3|4|5|6}[:SET]:CENTer . . . . . SPA 5-29
:CALCulate:MARKer{1|2|3|4|5|6}[:SET]:RLEVel . . . . . SPA 5-29

:CALCulate:MARKer{1|2|3|4|5|6}[:STATe] . . . . . SPA 5-29
:CONFigure:ACPower . . . . . SPA 5-30
:CONFigure:CHPower. . . . . SPA 5-30
:CONFigure:FSTRength . . . . . SPA 5-31
:CONFigure:OBWidth. . . . . SPA 5-31
:CONFigure? . . . . . SPA 5-32
:DISPlay:GRID . . . . . SPA 5-33
:DISPlay:WINDow:TRACe:Y:ADJust. . . . . SPA 5-33
:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision. . . . . SPA 5-33
:DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel. . . . . SPA 5-34
:DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel:OFFSet. . . . . SPA 5-34
:FETCh:ACPower?. . . . . SPA 5-35
:FETCh:CHPower:CHPower? . . . . . SPA 5-35
:FETCh:CHPower:DENSity? . . . . . SPA 5-36
:FETCh:CHPower? . . . . . SPA 5-36
:FETCh:OBWidth:FREQuency? . . . . . SPA 5-37
:FETCh:OBWidth? . . . . . SPA 5-37
:FORMat[:READings][:DATA] . . . . . SPA 5-38
:INITiate:CONTinuous . . . . . SPA 5-40
:INITiate[:IMMediate]. . . . . SPA 5-40
:INITiate:SAVe:ON:EVENt:CROSsing:LIMit OFF|ON|0|1 . . . . . SPA 5-41
:INITiate:SAVe:ON:EVENt:SWEep OFF|ON|0|1 . . . . . SPA 5-41
:INITiate:SAVe:THEn:STOp OFF|ON|0|1 . . . . . SPA 5-41
:MEASure:ACPower? . . . . . SPA 5-42
:MEASure:CHPower:CHPower?. . . . . SPA 5-43
:MEASure:CHPower:DENSity? . . . . . SPA 5-44
:MEASure:CHPower? . . . . . SPA 5-44
:MEASure:OBWidth? . . . . . SPA 5-45
:MMEMory:DATA? . . . . . SPA 5-46
:MMEMory:DELete . . . . . SPA 5-46

:MMEMory:LOAD:LIMit . . . . . SPA 5-47
:MMEMory:LOAD:STATe . . . . . SPA 5-47
:MMEMory:LOAD:TRACe . . . . . SPA 5-48
:MMEMory:STORe:LIMit. . . . . SPA 5-49
:MMEMory:STORe:STATe . . . . . SPA 5-49
:MMEMory:STORe:TRACe. . . . . SPA 5-50
:READ:ACPower? . . . . . SPA 5-51
:READ:CHPower:DENSity? . . . . . SPA 5-52
:READ:CHPower? . . . . . SPA 5-52
:READ:OBWidth? . . . . . SPA 5-53
:TRACe:COPY . . . . . SPA 5-54
:TRACe:EXCHange TRACE2,TRACE3 . . . . . SPA 5-54
:TRACe:PREamble? . . . . . SPA 5-55
:TRACe[:DATA]. . . . . SPA 5-62
:TRACe{1|2|3}:DISPlay[:STATe] . . . . . SPA 5-63
:TRACe{1|2|3}:WRITe[:STATe] . . . . . SPA 5-64
:TRACe1:OPERation. . . . . SPA 5-65
:TRACe2:OPERation. . . . . SPA 5-66
:TRACe3:OPERation. . . . . SPA 5-67
:TRIGger[:SEQuence]:SOURce . . . . . SPA 5-68
:TRIGger[:SEQuence]:VIDeo:LEVel . . . . . SPA 5-68
:TRIGger[:SEQuence]:VIDeo:POSition . . . . . SPA 5-69
:UNIT:POWer . . . . . SPA 5-70
[:SENSe]:ACPower:BANDwidth|BWIDth:ADJacent. . . . . SPA 5-71
[:SENSe]:ACPower:BANDwidth|BWIDth:MAIN. . . . . SPA 5-71
[:SENSe]:ACPower:BANDwidth|BWIDth:SPACing . . . . . SPA 5-72
[:SENSe]:ACPower:STATe . . . . . SPA 5-72
[:SENSe]:AVERage:COUNt . . . . . SPA 5-72
[:SENSe]:AVERage:TYPE . . . . . SPA 5-73
[:SENSe]:BANDwidth|BWIDth[:RESolution] . . . . . SPA 5-74

[:SENSe]:BANDwidth | BWIDth[:RESolution]:AUTO . . . . . SPA 5-74
[:SENSe]:BANDwidth | BWIDth[:RESolution]:RATio . . . . . SPA 5-75
[:SENSe]:BANDwidth | BWIDth:VIDeo . . . . . SPA 5-75
[:SENSe]:BANDwidth | BWIDth:VIDeo:AUTO . . . . . SPA 5-76
[:SENSe]:BANDwidth | BWIDth:VIDeo:RATio . . . . . SPA 5-76
[:SENSe]:CHPower:BANDwidth | BWIDth:INTegration . . . . . SPA 5-77
[:SENSe]:CHPower:STATe . . . . . SPA 5-77
[:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude] . . . . . SPA 5-78
[:SENSe]:CORRection:IMPedance[:INPut]:OFFSet . . . . . SPA 5-78
[:SENSe]:DETector[:FUNCtion] . . . . . SPA 5-79
[:SENSe]:EXTRefstatus? . . . . . SPA 5-79
[:SENSe]:FREQuency:CENTer . . . . . SPA 5-80
[:SENSe]:FREQuency:SIGStandard:CHANnel . . . . . SPA 5-80
[:SENSe]:FREQuency:SIGStandard:NAMe . . . . . SPA 5-81
[:SENSe]:FREQuency:SPAN . . . . . SPA 5-81
[:SENSe]:FREQuency:SPAN:FULL . . . . . SPA 5-82
[:SENSe]:FREQuency:SPAN:PREVious . . . . . SPA 5-82
[:SENSe]:FREQuency:STARt . . . . . SPA 5-82
[:SENSe]:FREQuency:STEP[:INCRement] . . . . . SPA 5-83
[:SENSe]:FREQuency:STOP . . . . . SPA 5-83
[:SENSe]:FSTRength:ANTenna . . . . . SPA 5-83
[:SENSe]:FSTRength:STATe . . . . . SPA 5-84
[:SENSe]:OBWidth:METHod . . . . . SPA 5-85
[:SENSe]:OBWidth:PERCent . . . . . SPA 5-85
[:SENSe]:OBWidth:STATe . . . . . SPA 5-86
[:SENSe]:OBWidth:XDB . . . . . SPA 5-86
[:SENSe]:POWer[:RF]:ATTenuation . . . . . SPA 5-87
[:SENSe]:POWer[:RF]:ATTenuation:AUTO . . . . . SPA 5-87
[:SENSe]:POWer[:RF]:GAIN[:STATe] . . . . . SPA 5-88
[:SENSe]:SWEep:STATus? . . . . . SPA 5-88

[:SENSe]:SWEep:TIME . . . . . SPA 5-88

[:SENSe]:SWEep:TRACe? . . . . . SPA 5-89

AM/FM/PM Commands 1

:CALCulate:MARKer{1|2|3|4|5|6}[:STATe] OFF|ON|0|1
:CALCulate:MARKer{1|2|3|4|5|6}[:STATe]? . . . . . AM/FM/PM 6-1

:CALCulate:MARKer{1|2|3|4|5|6}:DELTa[:STATe] OFF|ON|0|1
:CALCulate:MARKer{1|2|3|4|5|6}:DELTa[:STATe]? . . . . . AM/FM/PM 6-1

:CALCulate:MARKer{1|2|3|4|5|6}:DELTa:X <x-parameter>
:CALCulate:MARKer{1|2|3|4|5|6}:DELTa:X? . . . . . AM/FM/PM 6-2

:CALCulate:MARKer{1|2|3|4|5|6}:DELTa:Y? . . . . . AM/FM/PM 6-2

:CALCulate:MARKer{1|2|3|4|5|6}[:SET]:CENTer . . . . . AM/FM/PM 6-2

:CALCulate:MARKer{1|2|3|4|5|6}:MAXimum . . . . . AM/FM/PM 6-3

:CALCulate:MARKer{1|2|3|4|5|6}[:SET]:RLEVel . . . . . AM/FM/PM 6-3

:CALCulate:MARKer{1|2|3|4|5|6}:X <x-parameter>
:CALCulate:MARKer{1|2|3|4|5|6}:X? . . . . . AM/FM/PM 6-3

:CALCulate:MARKer{1|2|3|4|5|6}:Y? . . . . . AM/FM/PM 6-3

:CALCulate:MARKer:AOFF . . . . . AM/FM/PM 6-4

:CALCulate:MARKer:TABLe[:STATe] OFF|ON
:CALCulate:MARKer:TABLe[:STATe]? . . . . . AM/FM/PM 6-4

:DISPlay:WINDow:TRACe:Y:ADJust . . . . . AM/FM/PM 6-5

:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision <rel ampl>
:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision? . . . . . AM/FM/PM 6-5

:DISPlay:WINDow:TRACe:Y:AFPanalyzer:PWR:OFFSet <rel ampl>
:DISPlay:WINDow:TRACe:Y:AFPanalyzer:PWR:OFFSet? . . AM/FM/PM 6-5

:FORMat[:READings][:DATA] ASCii|INTeger,32|REAL,[<length>]
:FORMat[:READings][:DATA]? . . . . . AM/FM/PM 6-6

:INITiate[:IMMediate] . . . . . AM/FM/PM 6-8

:INITiate:CONTinuous OFF|ON|0|1
:INITiate:CONTinuous? . . . . . AM/FM/PM 6-8

:MMEMory:DELete <filename> . . . . . AM/FM/PM 6-9

:MMEMory:LOAD:STATe <integer>,<filename> . . . . . AM/FM/PM 6-9

:MMEMory:LOAD:TRACe <integer>,<filename> . . . . . AM/FM/PM 6-9

:MMEMory:STORe:STATe <integer>,<filename> . . . . . AM/FM/PM 6-10

:MMEMory:STORe:TRACe <integer>,<filename> . . . . . AM/FM/PM 6-11

:TRACe[:DATA]? . . . . . AM/FM/PM 6-12

[:SENSe]:AFPanalyzer:AVERage:COUNt <avg count>
[:SENSe]:AFPanalyzer:AVERage:COUNt? . . . . . AM/FM/PM 6-13

[:SENSe]:AFPanalyzer:DEMod:DATA? . . . . . AM/FM/PM 6-13

[:SENSe]:AFPanalyzer:DEMod:MODE RFSP | AFSP | AFWV | SUMMary
[:SENSe]:AFPanalyzer:DEMod:MODE? . . . . . AM/FM/PM 6-13

[:SENSe]:AFPanalyzer:DEMod:TYPE AM | FM | PM
[:SENSe]:AFPanalyzer:DEMod:TYPE? . . . . . AM/FM/PM 6-14

[:SENSe]:AFPanalyzer:FM:SCALe <percentage>
[:SENSe]:AFPanalyzer:FM:SCALe?. . . . . AM/FM/PM 6-14

[:SENSe]:AFPanalyzer:IFBW <freq>
[:SENSe]:AFPanalyzer:IFBW? . . . . . AM/FM/PM 6-14

[:SENSe]:AFPanalyzer:IFBW:AUTO 0 | 1
[:SENSe]:AFPanalyzer:IFBW:AUTO? . . . . . AM/FM/PM 6-15

[:SENSe]:AFPanalyzer:PM:SCALe <Radians>
[:SENSe]:AFPanalyzer:PM:SCALe?. . . . . AM/FM/PM 6-15

[:SENSe]:AFPanalyzer:RFSPectrum:DATA?. . . . . AM/FM/PM 6-15

[:SENSe]:AFPanalyzer:SPAN <freq>
[:SENSe]:AFPanalyzer:SPAN? . . . . . AM/FM/PM 6-16

[:SENSe]:AFPanalyzer:SWEep:TIME <time>
[:SENSe]:AFPanalyzer:SWEep:TIME? . . . . . AM/FM/PM 6-16

[:SENSe]:AFPanalyzer:SUMMary:DATA? . . . . . AM/FM/PM 6-16

[:SENSe]:FREQuency:CENTer <freq>
[:SENSe]:FREQuency:CENTer?. . . . . AM/FM/PM 6-17

[:SENSe]:FREQuency:SIGStandard:CHANnel <number>
[:SENSe]:FREQuency:SIGStandard:CHANnel? . . . . . AM/FM/PM 6-17

[:SENSe]:FREQuency:SIGStandard:NAMe <string>
[:SENSe]:FREQuency:SIGStandard:NAMe? . . . . . AM/FM/PM 6-17

[:SENSe]:FREQuency:SPAN <freq>
[:SENSe]:FREQuency:SPAN?. . . . . AM/FM/PM 6-18

[:SENSe]:FREQuency:SPAN:FULL . . . . . AM/FM/PM 6-18

[:SENSe]:FREQuency:SPAN:MINimum . . . . . AM/FM/PM 6-18

[:SENSe]:FREQuency:SPAN:PREVious . . . . . AM/FM/PM 6-18

[:SENSe]:FREQuency:STEP[:INCRement] <freq>
[:SENSe]:FREQuency:STEP[:INCRement]? . . . . . AM/FM/PM 6-18

[:SENSe]:OBWidth:METHod XDB | PERCent
[:SENSe]:OBWidth:METHod? . . . . . AM/FM/PM 6-19

[:SENSe]:OBWidth:PERCent <percentage>
[:SENSe]:OBWidth:PERCent? . . . . . AM/FM/PM 6-19

[:SENSe]:OBWidth:XDB <rel ampl>
[:SENSe]:OBWidth:XDB? . . . . . AM/FM/PM 6-20

All Mode Commands 1

:INSTrument:CATalog:FULL? . . . . . All Modes 7-2

:INSTrument:NSELect . . . . . All Modes 7-3

:INSTrument[:SELect] . . . . . All Modes 7-4

:MMEMory:DATA? . . . . . All Modes 7-6

:MMEMory:DELete. . . . . All Modes 7-6

MMEM:DIR? . . . . . All Modes 7-7

:MMEMory:MSIS . . . . . All Modes 7-8

:MMEMory:MSIS:COPY. . . . . All Modes 7-8

:MMEMory:MSIS:DESTination . . . . . All Modes 7-9

:MMEMory:STORe:JPEG. . . . . All Modes 7-10

:SYSTem:OPTions? . . . . . All Modes 7-11

:SYSTem:PRESet . . . . . All Modes 7-11

[:SENSe]:GPS . . . . . All Modes 7-12

[:SENSe]:GPS:RESet . . . . . All Modes 7-12

[:SENSe]:GPS:CURRent? . . . . . All Modes 7-12

[:SENSe]:GPS:VOLTage. . . . . All Modes 7-13

:FETCh:GPS? . . . . . All Modes 7-14

:SYSTem:WINDow:CLOSe. . . . . All Modes 7-14

10580-00302

L

Anritsu utilizes recycled paper and environmentally conscious inks and toner.

Anritsu Company
490 Jarvis Drive
Morgan Hill, CA 95037-2809
USA
http://www.anritsu.com