SSA3000X Plus
Spectrum Analyzer

SSA3000X-R
Real time spectrum Analyzer

SVA1000X
Spectrum Analyzer

Programming Guide

PG0703P_E02A

SIGLENT TECHNOLOGIES CO.,LTD

Contents

1.Programming Overview

The Siglent SVA1000X series spectrum analyzers features LAN, USB Device, and SIGLENT GPIB_USB module interfaces. By using a computer with these interfaces, and a suitable programming language (and/or NI-VISA software), users can remotely control the analyzer based on SCPI (Standard Commands for Programmable Instruments) command set, Labview and IVI (Interchangeable Virtual Instrument), to interoperate with other programmable instruments.

This chapter introduces how to build communication between the spectrum analyzer and a controller computer with these interfaces.

1.1 Remotely Operating the Analyzer

The analyzer provides both the USB and LAN connection which allows you to set up a remote operation environment with a controller computer. A controller computer could be a personal computer (PC) or a minicomputer. Some intelligent instruments also function as controllers.

1.1.1 USB: Connecting the Analyzer via the USB Device port

Refer to the following steps to finish the connection via USB-Device:

1. Install NI-VISA on your PC for USB-TMC driver.
2. Connect the analyzer USB Device port to a PC with a USB A-B cable.

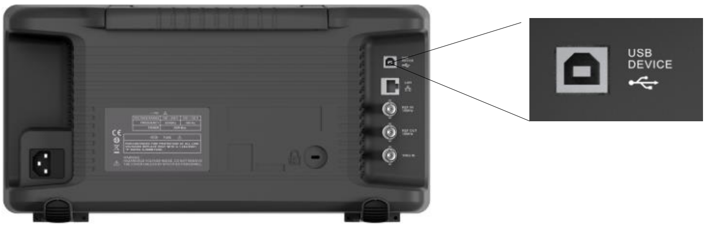


Figure 1-1 USB Device

3. Switch on the analyzer.

The analyzer will be detected automatically as a new USB hardware.

1.1.2 LAN: Connecting the Analyzer via the LAN port

Refer to the following steps to finish the connection via LAN:

1. Install NI-VISA on your PC for VXI driver. Or without NI-VISA, using socket in your PC's Operating System.
2. Connect the analyzer to PC or the local area network with a LAN cable.

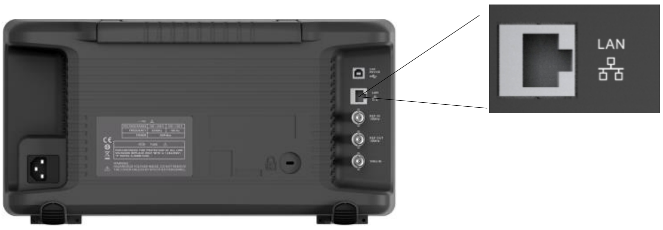


Figure 1-2 LAN

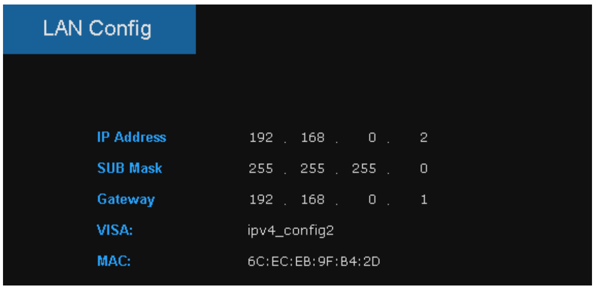
3.Switch on the analyzer.

4.Press button on the front panel **System** →Interface→LAN to enter the LAN Config function menu.

5.Select the IP Config between Static and DHCP.

◆DHCP: the DHCP server in the current network will assign the network parameters automatically (IP address, subnet mask, gate way) for the analyzer.

◆Static: you can set the IP address, subnet mask, gate way manually. Press Apply.



LAN Config

IP Address	192 . 168 . 0 . 2
SUB Mask	255 . 255 . 255 . 0
Gateway	192 . 168 . 0 . 1
VISA:	ipv4_config2
MAC:	6C:EC:EB:9F:B4:2D

Figure 1-3 LAN Config

The analyzer will be detected automatically or manually as a new LAN point.

1.1.3 GPIB: Connecting the Analyzer via the USB Host port

Refer to the following steps to finish the connection via USB:

1.Install NI-VISA on your PC for GPIB driver.

2.Connect the analyzer USB Host port to a PC’s GPIB card port, with SIGLENT USB-GPIB adaptor.

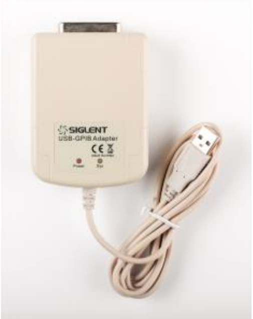


Figure 1-4 SIGLENT USB-GPIB Adaptor

3.Switch on the analyzer.

4.Press button on the front panel **System**→Interface→GPIB to enter the GPIB number.

The analyzer will be detected automatically as a new GPIB point.

1.2 Build Communication

1.2.1 Build Communication Using VISA

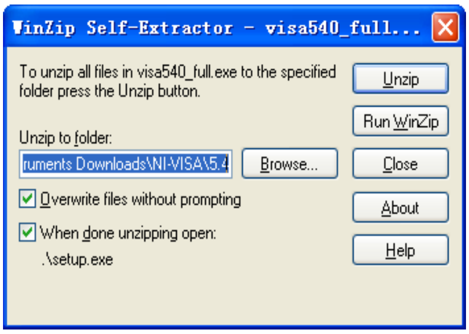
NI-VISA includes a Run-Time Engine version and a Full version. The Run-Time Engine version provides NI device drivers such as USB-TMC, VXI, GPIB, etc. The full version includes the Run-Time Engine and a software tool named NI MAX that provides a user interface to control the device.

You can get NI-VISA full version from:

http://www.ni.com/download/.

After download you can follow the steps below to install it:

a.Double click the visa_full.exe, dialog shown as below:



WinZip Self-Extractor - visa540_full...

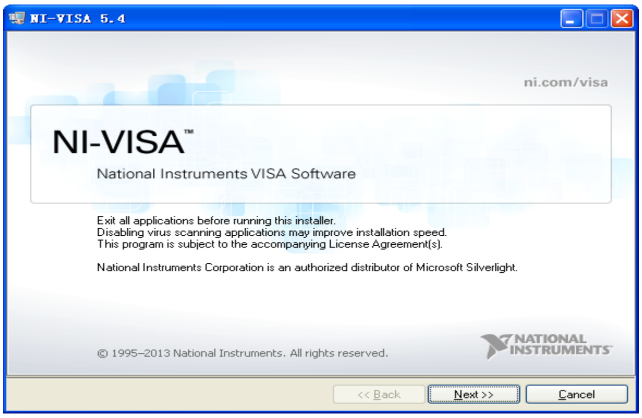
To unzip all files in visa540_full.exe to the specified folder press the Unzip button.

Unzip to folder: ruments Downloads\NI-VISA\5.4

☑ Overwrite files without prompting

☑ When done unzipping open: .\setup.exe

b.Click Unzip, the installation process will automatically launch after unzipping files. If your computer needs to install .NET Framework 4, its setup process will auto start.



NI-VISA 5.4

ni.com/visa

NI-VISA™

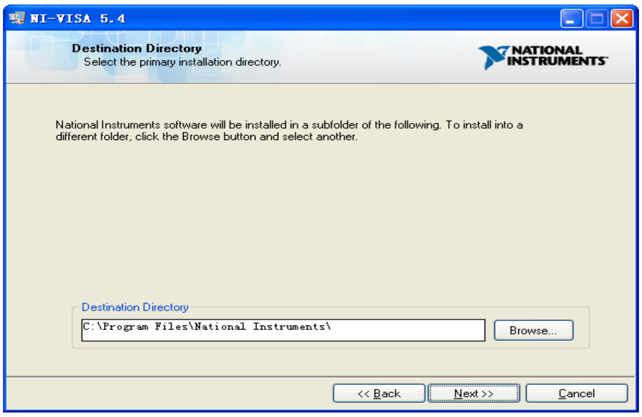
National Instruments VISA Software

Exit all applications before running this installer.
Disabling virus scanning applications may improve installation speed.
This program is subject to the accompanying License Agreement(s).

National Instruments Corporation is an authorized distributor of Microsoft Silverlight.

© 1995–2013 National Instruments. All rights reserved.

c.The NI-VISA installing dialog is shown above. Click Next to start the installation process.

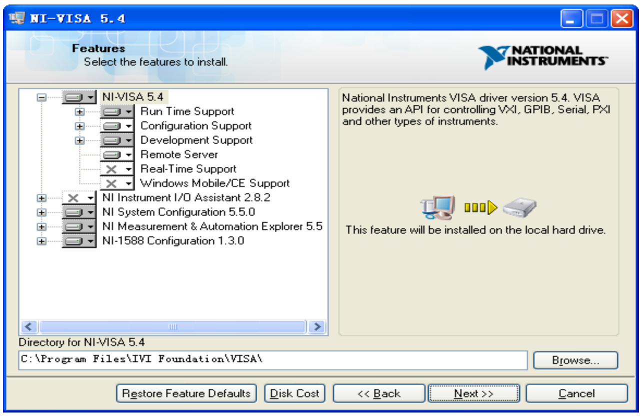


Destination Directory
Select the primary installation directory.

National Instruments software will be installed in a subfolder of the following. To install into a different folder, click the Browse button and select another.

Destination Directory
C:\Program Files\National Instruments\

Set the install path, default path is “C:\Program Files\National Instruments\”, you can change it. Click Next, dialog shown as above.



Features
Select the features to install.

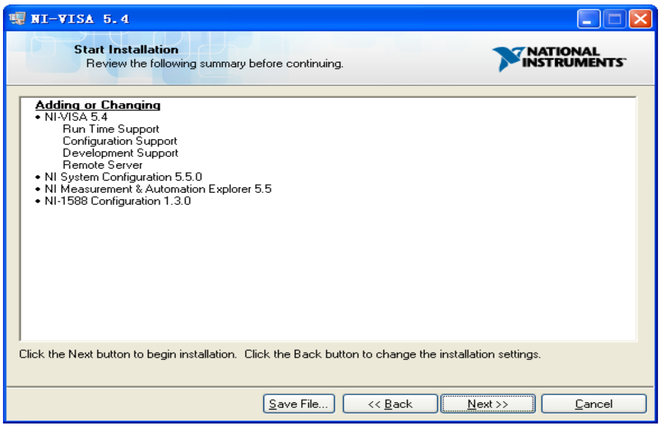
- NI-VISA 5.4
  - Run Time Support
  - Configuration Support
  - Development Support
  - Remote Server
  - Real-Time Support
  - Windows Mobile/CE Support
- NI Instrument I/O Assistant 2.8.2
- NI System Configuration 5.5.0
- NI Measurement & Automation Explorer 5.5
- NI-1588 Configuration 1.3.0

National Instruments VISA driver version 5.4. VISA provides an API for controlling VXI, GPIB, Serial, PXI and other types of instruments.

This feature will be installed on the local hard drive.

Directory for NI-VISA 5.4
C:\Program Files\IVI Foundation\VISA\

d.Click Next twice, in the License Agreement dialog, select the “ I accept the above 2 License Agreement(s).” ,and click Next, dialog shown as below:

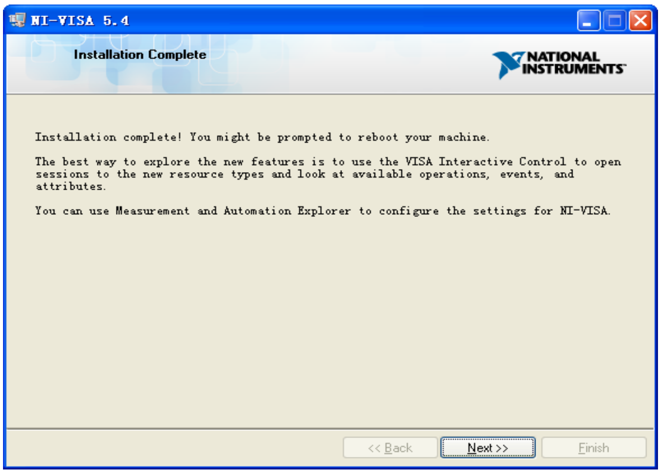


Start Installation
Review the following summary before continuing.

Adding or Changing
- NI-VISA 5.4
  - Run Time Support
  - Configuration Support
  - Development Support
  - Remote Server
- NI System Configuration 5.5.0
- NI Measurement & Automation Explorer 5.5
- NI-1588 Configuration 1.3.0

Click the Next button to begin installation. Click the Back button to change the installation settings.

e.Click Next to run installation.



Installation complete! You might be prompted to reboot your machine.

The best way to explore the new features is to use the VISA Interactive Control to open sessions to the new resource types and look at available operations, events, and attributes.

You can use Measurement and Automation Explorer to configure the settings for NI-VISA.

Now the installation is complete, reboot your PC.

1.2.2 Build Communication Using Sockets

Through the LAN interface, VXI-11, Sockets and Telnet protocols can be used to communicate with the spectrum analyzer. VXI-11 is provided in NI-VISA, while Sockets and Telnet are commonly included in PC's OS initially.

Socket LAN is a method used to communicate with the spectrum analyzer over the LAN interface using the Transmission Control Protocol/Internet Protocol (TCP/IP). A socket is a fundamental technology used for computer networking and allows applications to communicate using standard mechanisms built into network hardware and operating systems. The method accesses a port on the spectrum analyzer from which bidirectional communication with a network computer can be established.

Before you can use sockets LAN, you must select the analyzer's sockets port number to use:

◆Standard mode. Available on port 5025. Use this port for programming.

1.3 Remote Control Capabilities

1.3.1 User-defined Programming

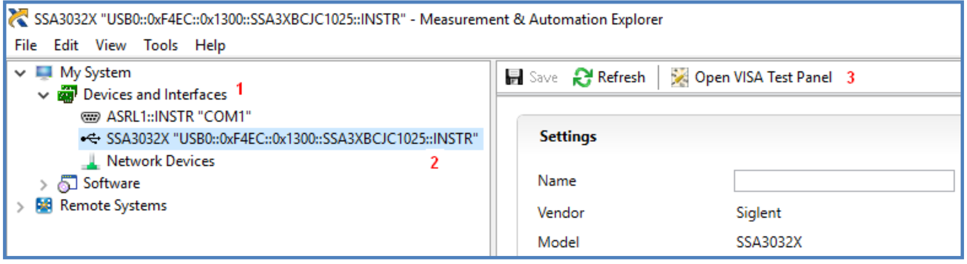
Users can use SCPI commands to program and control the spectrum analyzer. For details, refer to the introductions in “**Programming Examples**”.

1.3.2 Send SCPI Commands via NI MAX

Users can control the spectrum analyzer remotely by sending SCPI commands via NI-MAX software. NI-MAX is National Instruments Measurement and Automation Explorer. It is an executable program that enables easy communication to troubleshoot issues with instrumentation.

1.3.2.1 Using USB

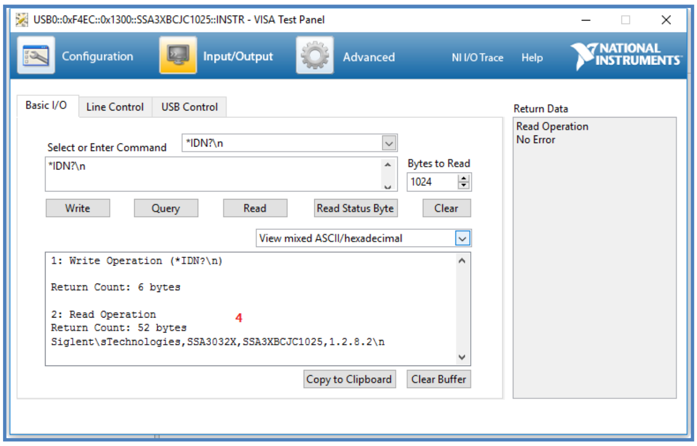
Run NI MAX software.
1.Click “Device and Interface” at the upper left corner of the software;
2. Find the “USBTMC” device symbol.



SSA3032X "USB0::0xF4EC::0x1300::SSA3XBCJC1025::INSTR" - Measurement & Automation Explorer

Name	
Vendor	Siglent
Model	SSA3032X

3. Click “Open VISA Test Panel” option button, then the following interface will appear.
4.Click the “Input/Output” option button and click the “Query” option button in order to view the operation information.



USB0::0xF4EC::0x1300::SSA3XBCJC1025::INSTR - VISA Test Panel

Select or Enter Command: *IDN?\n

*IDN?\n

Bytes to Read: 1024

View mixed ASCII/hexadecimal

```
1: Write Operation (*IDN?\n)

Return Count: 6 bytes

2: Read Operation
Return Count: 52 bytes
Siglent\sTechnologies,SSA3032X,SSA3XBCJC1025,1.2.8.2\n
```

Return Data

Read Operation
No Error

NOTE: The *IDN? command (known as the Identification Query) returns the instrument manufacturer, instrument model, serial number, and other identification information.

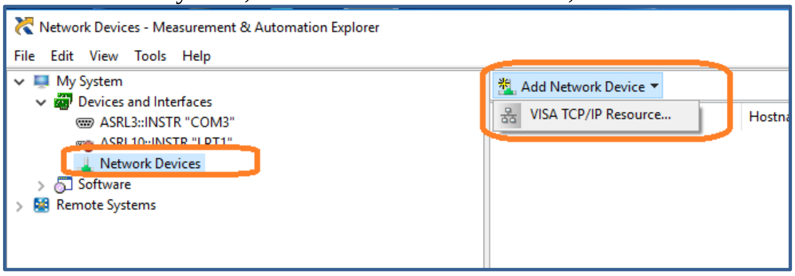
1.3.2.2 Using LAN

Select, Add Network Device, and select VISA TCP/IP Resource as shown:

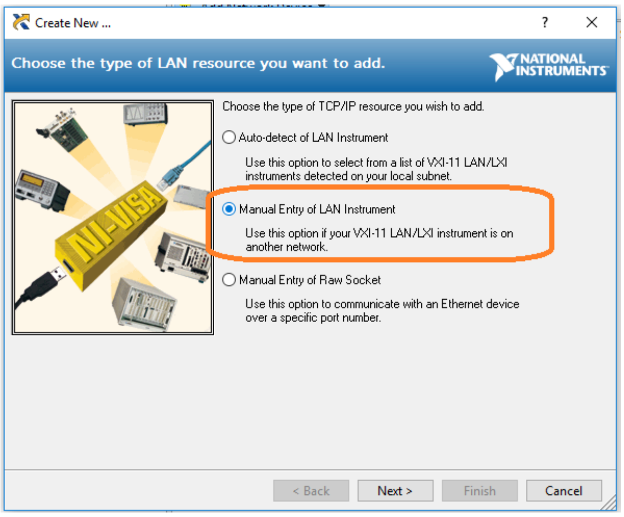
Run NI MAX software.

1.Click “Device and Interface” at the upper left corner of the software;

2.Find the “Network Devices” symbol, click “Add Network Device”;



3.Select Manual Entry of LAN instrument, select Next, and enter the IP address as shown. Click Finish to establish the connection:



Choose the type of LAN resource you want to add.

Choose the type of TCP/IP resource you wish to add.

○ Auto-detect of LAN Instrument

Use this option to select from a list of VXI-11 LAN/LXI instruments detected on your local subnet.

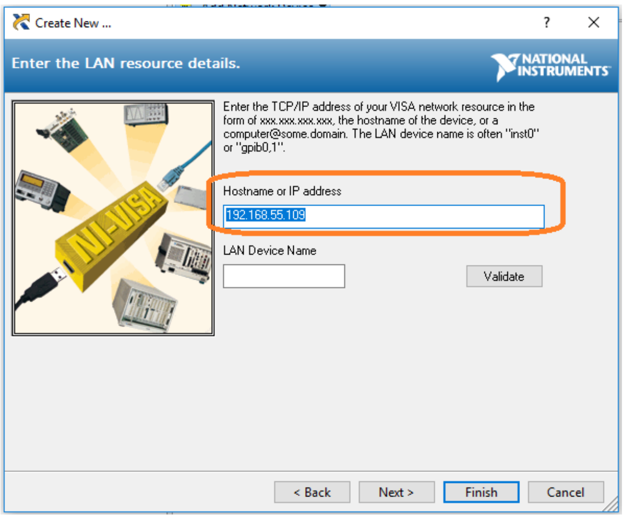
● Manual Entry of LAN Instrument

Use this option if your VXI-11 LAN/LXI instrument is on another network.

○ Manual Entry of Raw Socket

Use this option to communicate with an Ethernet device over a specific port number.

NOTE: Leave the LAN Device Name BLANK or the connection will fail.



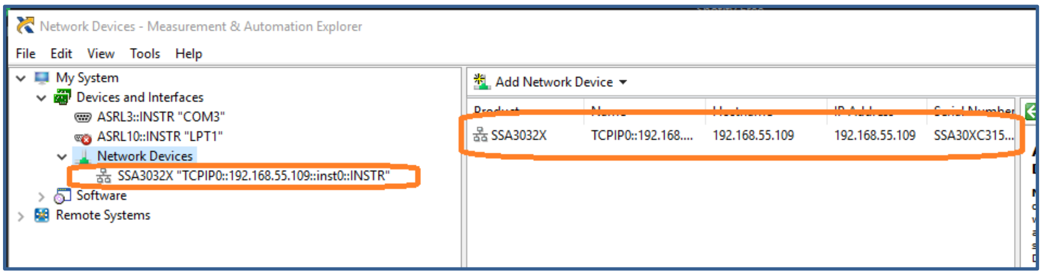
Enter the LAN resource details.

Enter the TCP/IP address of your VISA network resource in the form of xxx.xxx.xxx.xxx, the hostname of the device, or a computer@some.domain. The LAN device name is often "inst0" or "gpib0,1".

Hostname or IP address: 192.168.55.109

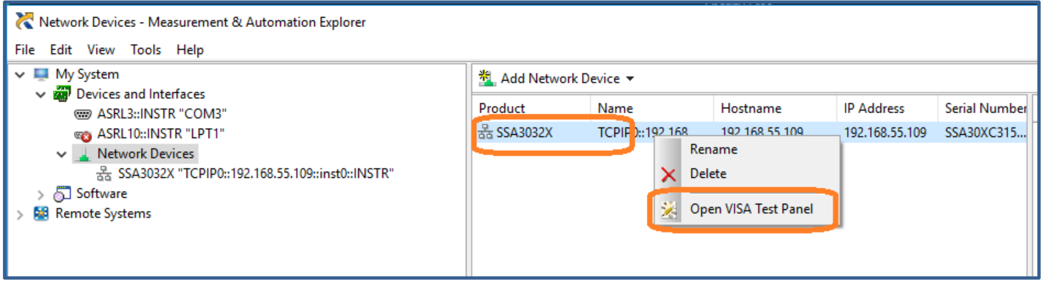
LAN Device Name:

4. After a brief scan, the connection should be shown under Network Devices:

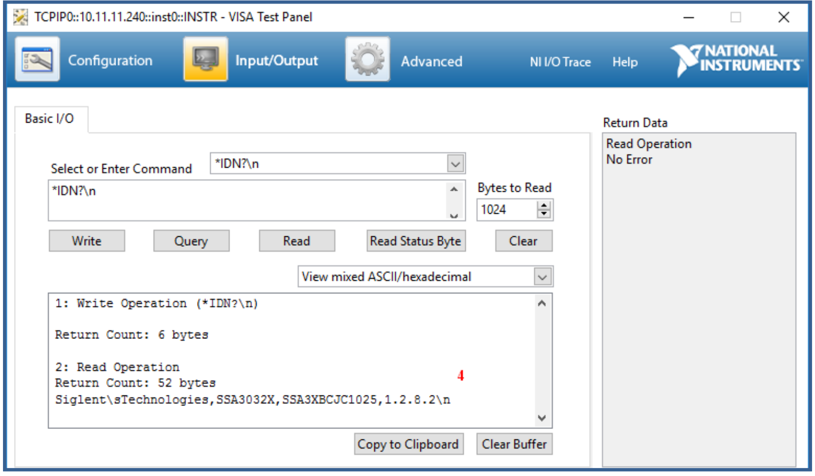


Product	Name	Hostname	IP Address	Serial Number
SSA3032X	TCPIP0::192.168....	192.168.55.109	192.168.55.109	SSA30XC315...

5.Right-click on the product and select Open NI-VISA Test Panel:



6.Click “Input/Output” option button and click “Query” option button. If everything is OK, you will see the Read operation information returned as shown below.



Select or Enter Command: *IDN?\n

Bytes to Read: 1024

```
1: Write Operation (*IDN?\n)

Return Count: 6 bytes

2: Read Operation
Return Count: 52 bytes
Siglent\sTechnologies,SSA3032X,SSA3XBCJC1025,1.2.8.2\n
```

Return Data: Read Operation No Error

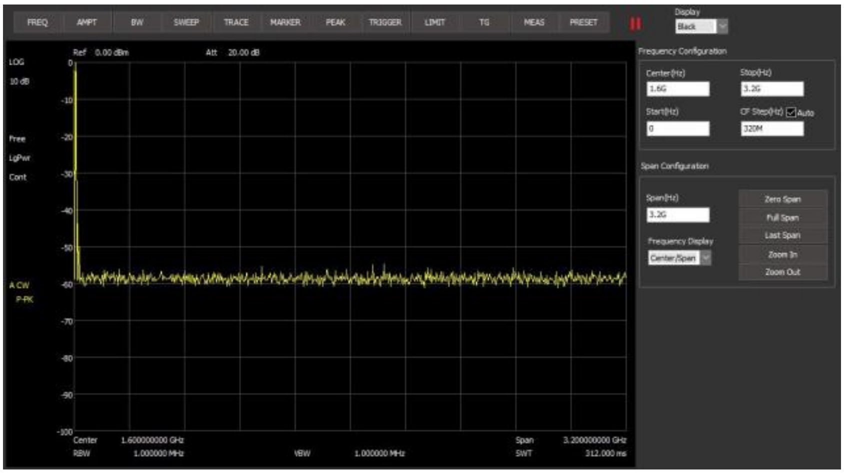
1.3.3 EasySpectrum Software

Users can control the spectrum analyzer remotely by EasySpectrum. PC software EasySpectrum is an easy-to-use, PC-Windows-based remote control tool for Siglent’s spectrum analyzer. You can download it from Siglent’s website. To connect the analyzer via the USB/LAN port to a PC, you need install the NI VISA first.

It is able to be used as:

- ◆A monitor to display and control the trace scans simultaneously with the analyzer;
- ◆A file maker to get user defined Limit/Correction files, and load them to the analyzer;
- ◆An EMI receiver to perform EMI Pre-compliance test including prescan, peak search, finalscan and report generating.

For the further description of the software, please refer to the online help embedded in this software.



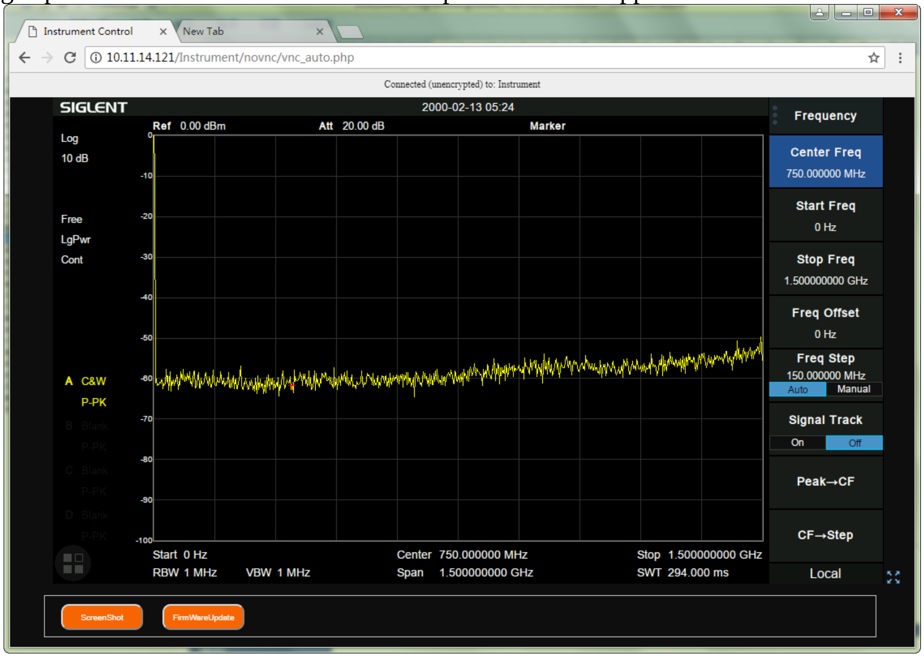
Ref 0.00 dBm Att 20.00 dB

Center(Hz) 1.6G
Stop(Hz) 3.2G
Start(Hz) 0
CF Step(Hz) 320M
Span(Hz) 3.2G
Frequency Display Center/Span

Center 1.600000000 GHz Span 3.200000000 GHz
RBW 1.000000 MHz VBW 1.000000 MHz SWT 312.000 ms

1.3.4 Web Control

With the embedded web server, the analyzer can be controlled through LAN from a web browser* on PC and mobile terminals, without any extra driver be installed. This provides remote controlling and monitoring capabilities. Screenshot and firmware update are also supported.



10.11.14.121/Instrument/novnc/vnc_auto.php

Connected (unencrypted) to: Instrument

2000-02-13 05:24

Ref 0.00 dBm Att 20.00 dB

Center Freq 750.000000 MHz
Start Freq 0 Hz
Stop Freq 1.500000000 GHz
Freq Offset 0 Hz
Freq Step 150.000000 MHz

Start 0 Hz Center 750.000000 MHz Stop 1.500000000 GHz
RBW 1 MHz VBW 1 MHz Span 1.500000000 GHz SWT 294.000 ms

*Web browser with HTML5 supported like Google Chrome or Firefox are recommended.

2.SCPI Overview

2.1 Command Format

SCPI commands present a hierarchical tree structure containing multiple subsystems, each of the subsystems is made up of a root keyword and several subkeywords. The command string usually starts with “:”, the keywords are separated by “:” and the followed parameter settings are separated by space. Query commands add “?” at the end of the string.

For example:

:SENSe:FREQuency:CENTer <freq>

:SENSe:FREQuency:CENTer?

SENSe is the root key of the command, FREQuency and CENTer are second and third keywords. The command begins with “**:**”, and separates the keywords at the same time, <freq> separated by space and represents the parameter available for setting; “**?**” represents a query.

2.2 Symbol Instruction

The following four symbols are not the content of SCPI commands and cannot be sent with the commands, but are usually used in the commands.

1.Triangle Brackets < >

The parameter in the triangle brackets must be replaced by an effective value. For example:

Send the “:DEMod:VOLume <value>” command in “:DEMod:VOLume 5”.

2.Square Brackets [ ]

The content in the square brackets can be ignored. When the parameter is ignored, the instrument will set the parameter to its default. For example,

In the “[:SENSe]:POWer[:RF]:ATTenuation?” command, sending any of the four commands below can generate the same effect:

:POWer:ATTenuation?

:POWer:RF:ATTenuation?

:SENSe:POWer:ATTenuation?

:SENSe:POWer:RF:ATTenuation?

3.Vertical Bar |

The vertical bar is used to separate multiple parameters and when sending the command, you can choose one of the parameters. For example,

In the “[:SENSe]:FREQuency:CENTer:STEP:AUTO OFF|ON|0|1” command, the parameters available are “OFF”, “ON”, “0” or “1”.

4.Braces { }

The parameters in the braces are optional which can be ignored or set for one or more times. For example:

:CALCulate:LLINe[1]|2:DATA <x-axis>,<ampl>{,<x-axis>, <ampl>}, in the command, the {,<x-axis>, <ampl>} parameters can be ignored or set for one or more times.

2.3 Parameter Type

The parameters in the commands introduced in this manual include 6 types: boolean, enumeration, integer, float, discrete and string.

1. Boolean

The parameters in the commands could be “OFF”, “ON”, “0” or “1”. For example:

[:SENSe]:FREQuency:CENTer:STEP:AUTO OFF|ON|0|1

2.Enumeration

The parameter could be any of the values listed. For example:

[:SENSe]:AVERage:TYPE LOGPower|POWer|VOLTage

The parameter is “OGPower”, “POWer” or “VOLTage”.

3.String

The parameter should be the combinations of ASCII characters. For example:

:SYSTem:COMMunicate:LAN:IPADdress <“xxx.xxx.xxx.xxx”>

The parameter can be set as “192.168.1.12” string.

4.Integer

Except other notes, the parameter can be any integer within the effective value range. For example:

[:SENSe]:DEMod:VOLume <value>

The parameter < value > can be set to any integer between 0 and 10.

5.Float

The parameter could be any value within the effective value range according to the accuracy requirement (the default accuracy contains up to 9 digits after the decimal points). For example:

:CALCulate:BANDwidth:NDB <value>

The parameter < value > can be set to any real number between -100 and 100.

6.Discrete

The parameter could only be one of the specified values and these values are discontinuous. For example:

[:SENSe]:BWIDth:VIDeo:RATio <number>

The parameter <number> could only be one of 0.001, 0.003, 0.01, 0.03, 0.1, 0.3, 1.0, 3.0, 10.0, 30.0, 100.0, 300.0, 1000.0.

2.4 Command Abbreviation

All of the commands are not case sensitive, so you can use any of them. But if abbreviation is used, all the capital letters in the command must be written completely. For example:

:DISPlay:WINDow:TRACe:Y:DLINe:STATe?

Can be abbreviated to:

:DISP:WIND:TRAC:Y:DLIN:STAT?

3.Commands that are Common to All Modes

3.1 IEEE Common Commands

*IDN

*RST

*CLS

*ESE

*ESR?

*OPC

*SRE

*STB?

*WAI

*TRG

*TST?

Command Format	*IDN?
Instruction	Returns an instrument identification information string. The string will contain the manufacturer, model number, serial number, software number, FPGA number and CPLD number.
Menu	None
Example	*IDN? Return: Siglent,SVA1015,1234567890,100.01.01.06.01

Command Format	*RST
Instruction	This command presets the instrument to a factory defined condition that is appropriate for remote programming operation.

Menu	None
Example	*RST

Command Format	***CLS**
Instruction	Clears the status byte register. It does this by emptying the error queue and clearing all bits in all of the event registers. The status byte register summarizes the states of the other registers. It is also responsible for generating service requests.
Menu	None
Example	*CLS

Command Format	***ESE <number>** ***ESE?**
Instruction	Set the bits in the standard event status enable register. This register monitors I/O errors and synchronization conditions such as operation complete, request control, query error, device dependent error, execution error, command error and power on. A summary bit is generated on execution of the command. The query returns the state of the standard event status enable register.
Menu	None
Example	*ESE 16

Command Format	***ESR?**
Instruction	Queries and clears the standard event status event register. (This is a destructive read.) The value returned reflects the current state (0/1) of all the bits in the register.
Menu	None
Example	*ESR?

Command Format	***OPC** ***OPC?**
Instruction	Set bit 0 in the standard event status register to “1” when all pending operations have finished. The query stops any new commands from being processed until the current processing is complete. Then it returns a “1”, and the program continues. This query can be used to synchronize events of other instruments on the external bus. Returns a “1” if the last processing is complete. Use this query when there’s a need to monitor the command execution status, such as a sweep execution.
Menu	None
Example	*OPC?

Command Format	***SRE <integer>** ***SRE?**

Instruction	This command enables the desired bits of the service request enable register. The query returns the value of the register, indicating which bits are currently enabled. The default value is 255.
Menu	None
Example	*SRE 1

Command Format	***STB?**
Instruction	This query is used by some instruments for a self test.
Menu	None
Example	*STB?

Command Format	***WAI**
Instruction	This command causes the instrument to wait until all pending commands are completed before executing any additional commands. There is no query form to the command.
Menu	None
Example	*WAI

Command Format	***TRG**
Instruction	Restarts the current sweep.
Menu	None
Example	*TRG

Command Format	***TST?**
Instruction	This query is used by some instruments for a self test.
Menu	None
Example	*TST?

3.2 System Subsystem

:SYSTem:TIME

:SYSTem:DATE

:SYSTem:COMMunicate:LAN:IPADdress

:SYSTem:COMMunicate:LAN:GATeway

:SYSTem:COMMunicate:LAN:SMASk

:SYSTem:COMMunicate:LAN:TYPE

:SYSTem:LANGuage

:SYSTem:PON:TYPE

:SYSTem:RESTart

:SYSTem:PRESet

:SYSTem:PRESet:TYPE

:SYSTem:PRESet:USER[1]|2|3|4|5|6|7:SAVE

:SYSTem:PRESet:USER[1]|2|3|4|5|6|7:LOAD

:SYSTem:FDEFault

:SYSTem:LKEY

:SYSTem:OPTions?

:SYSTem:POWer:OFF

:SYSTem:CONFigure:SYSTem?

Command Format	**:SYSTem:TIME <hhmmss>** **:SYSTem:TIME?**
Instruction	Sets System time. Gets System time.
Parameter Type	String
Parameter Range	hour(0~23), minute(0~59), second(0~59)
Return	String
Default	None
Menu	System > Date & Time
Example	Sets System time: :SYSTem:TIME 182559 Gets System time: :SYSTem:TIME?

Command Format	**:SYSTem:DATE <yyyymmdd>** **:SYSTem:DATE?**
Instruction	Sets system date. Gets system date.
Parameter Type	String
Parameter Range	year(four digits), month(1~12), date(1~31)
Return	String
Default	None
Menu	System > Date & Time

Example	Sets System date: :SYSTem:DATE 20050101 Gets System date: :SYSTem:DATE?

Command Format	**:SYSTem:COMMunicate:LAN:IPADdress <“xxx.xxx.xxx.xxx”> :SYSTem:COMMunicate:LAN:IPADdress?**
Instruction	Sets a host name for the analyzer in network. IP Address command will be effective after using this “APPLy” command. Gets IP address.
Parameter Type	String
Parameter Range	Conform to the IP Sets standard(0-255:0-255:0-255:0-255)
Return	IP address String
Default	None
Menu	System > Interface > LAN > IP Address
Example	:SYSTem:COMMunicate:LAN:IPADdress “192.168.1.12” :SYSTem:COMMunicate:LAN:IPADdress?

Command Format	**:SYSTem:COMMunicate:LAN:GATeway <“xxx.xxx.xxx.xxx”> :SYSTem:COMMunicate:LAN:GATeway?**
Instruction	Sets the gateway for the analyzer in the network. The gateway will be fetched automatically if the IP assignment is set to DHCP. Gateway command will be effective after using this “APPLy” command. Gets gateway.
Parameter Type	String
Parameter Range	Conform to the IP standard (0-255:0-255:0-255:0-255)
Return	Gateway string.
Default	None
Menu	System > Interface > LAN > Gateway
Example	:SYSTem:COMMunicate:LAN:GATeway “192.168.1.1” :SYSTem:COMMunicate:LAN:GATeway?

Command Format	**:SYSTem:COMMunicate:LAN:SMASk <“xxx.xxx.xxx.xxx”> :SYSTem:COMMunicate:LAN:SMASk?**
Instruction	Sets the subnet mask according to the PC network Settings. The subnet mask will be set automatically if the IP assignment is set to DHCP. Subnet Mask commands will be effective after using this “APPLy” command. Gets Subnet Mask.
Parameter Type	String
Parameter Range	Conform to the IP standard (0-255:0-255:0-255:0-255)
Return	Subnet mask string
Default	None

Menu	System > Interface > LAN > Subnet Mask
Example	:SYSTem:COMMunicate:LAN:SMASk?

Command Format	**:SYSTem:COMMunicate:LAN:TYPE STATIC|DHCP** **:SYSTem:COMMunicate:LAN:TYPE?**
Instruction	Toggles the IP assignment Setting between static (manual) and DHCP (dynamic assignment) mode. Gets IP config.
Parameter Type	Enumeration
Parameter Range	STATIC|DHCP
Return	Enumeration
Default	None
Menu	System > Interface > LAN > IP Config
Example	:SYSTem:COMMunicate:LAN:TYPE DHCP :SYSTem:COMMunicate:LAN:TYPE?

Command Format	**:SYSTem:LANGuage SCHINESE|ENGLISH** **:SYSTem:LANGuage?**
Instruction	Sets language. Gets language.
Parameter Type	Enumeration
Parameter Range	SCHINESE: Chinese ENGLISH: English
Return	Enumeration
Default	None
Menu	System > Language
Example	Sets language :SYSTem:LANGuage SCHINESE Gets language :SYSTem:LANGuage?

Command Format	**:SYSTem:PON:TYPE DFT|LAST|USER** **:SYSTem:PON:TYPE?**
Instruction	Uses command to set analyzer to power on in default, user, or last state. Gets power on type.
Parameter Type	Enumeration
Parameter Range	DFT: Default LAST: Last USER: Custom Configuration
Return	Enumeration
Default	DFT
Menu	System > Pwr On/Preset > Power On

Example	SYSTem:PON:TYPE DFT

Command Format	**:SYSTem:PRESet**
Instruction	Use this command to preset the instrument. The preset type is based on the Setting of Preset Type: DFT, User or Last.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	None
Example	:SYSTem:PRESet

Command Format	**:SYSTem:RESTart**
Instruction	Use this command to restart the instrument (part of machine may not support).
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	None
Example	:SYSTem:RESTart

Command Format	**:SYSTem:PRESet:TYPE DFT|LAST|USER** **:SYSTem:PRESet:TYPE?**
Instruction	Uses this command to preset the analyzer to default, user, or last state. Gets preset type.
Parameter Type	Enumeration
Parameter Range	DFT: Default LAST: Last USER: Custom Configuration
Return	Enumeration
Default	DFT
Menu	System > Pwr On/Preset > Preset
Example	:SYSTem:PRESet:TYPE DFT

Command Format	:SYSTem:PRESet:USER[1]|2|3|4|5|6|7:SAVE
Instruction	Saves current setting to user config.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	System > Pwr/Preset > User Config
Example	:SYSTem:PRESet:USER7:SAVE

Command Format	:SYSTem:PRESet:USER[1]|2|3|4|5|6|7:LOAD
Instruction	Loads user config.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	System > Pwr On/Preset > User Config
Example	:SYSTem:PRESet:USER6:LOAD

Command Format	:SYSTem:FDEFault
Instruction	Sets both the measure and setting parameters to factory preset parameters.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	System > Pwr On/Preset > Factory Reset
Example	:SYSTem:FDEFault

Command Format	:SYSTem:LKEY <“option”>,<“license key”>
Instruction	Uses this command to enable the specified option with the license key, please restart the instrument to make license active.
Parameter Type	“option”: Enumeration “license key”: String

Parameter Range	“option”: Meas|EMI|TG|DMA|DTF|VNA “license key”: provided by Siglent Technologies, 16 bits String.
Return	None
Default	None
Menu	System > System Info > Load Option
Example	:SYSTem:LKEY EMI,fjbdajffnklmgwno

Command Format	**:SYSTem:OPTions?**
Instruction	This command returns a list of the options that are installed.
Parameter Type	None
Parameter Range	None
Return	Meas|EMI|TG|DMA|DTF|VNA
Default	None
Menu	System > System Info
Example	:SYSTem:OPTions?

Command Format	**:SYSTem:POWer:OFF**
Instruction	Uses this command to turn off the instrument.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	None
Example	:SYSTem:POWer:OFF

Command Format	**:SYSTem:CONFigure:SYSTem?**
Instruction	Uses this command to query the system message of the instrument.
Parameter Type	None
Parameter Range	None
Return	String
Default	None
Menu	System > System Info

Example	:SYSTem:CONFigure:SYSTem?

3.3 Memory Subsystem

:MMEMory:STORe

:MMEMory:LOAD

:MMEMory:DELete

Command Format	**:MMEMory:STORe STA|TRC|COR|CSV|LIM|JPG|BMP|PNG, ”<file>”**
Instruction	Stores file.
Parameter Type	String
Parameter Range	None
Return	None
Default	None
Menu	File > Save
Example	:MMEMory:STORe STA,”ABC.sta”

Command Format	**:MMEMory:LOAD STA|TRC|COR|LIM, ”<file>”**
Instruction	Loads file.
Parameter Type	String
Parameter Range	None
Return	None
Default	None
Menu	File > Open/Load
Example	:MMEMory:LOAD STA, ”ABC.sta”

Command Format	**:MMEMory:DELete ”<file>”**
Instruction	Deletes file or folder.
Parameter Type	String
Parameter Range	None
Return	None
Default	None

Menu	File > Operate > Delete
Example	:MMEMory:DELete ”ABC.sta”

3.4 Display Subsection

:DISPlay:WINDow:TRACe:GRATicule:GRID:BRIGhtness

:DISPlay:WINDow:TRACe:Y:DLINe:STATe

:DISPlay:WINDow:TRACe:Y:DLINe

Command Format	**:DISPlay:WINDow:TRACe:GRATicule:GRID:BRIGhtness <value>** **:DISPlay:WINDow:TRACe:GRATicule:GRID:BRIGhtness?**
Instruction	Sets grid brightness. Gets grid brightness.
Parameter Type	Integer
Parameter Range	0 ~ 100
Return	Integer
Default	30%
Menu	Display > Grid Brightness
Example	:DISPlay:WINDow:TRACe:GRATicule:GRID:BRIGhtness 50

Command Format	**:DISPlay:WINDow:TRACe:Y:DLINe:STATe OFF|ON|0|1** **:DISPlay:WINDow:TRACe:Y:DLINe:STATe?**
Instruction	Toggles the display line between on and off. Gets the display line state.
Parameter Type	Enumeration
Parameter Range	OFF|ON|0|1
Return	0|1
Default	OFF
Menu	Display > Display Line
Example	:DISPlay:WINDow:TRACe:Y:DLINe:STATe ON

Command Format	**:DISPlay:WINDow:TRACe:Y:DLINe <value>** **:DISPlay:WINDow:TRACe:Y:DLINe?**
Instruction	Sets the amplitude value for the display line. Gets the amplitude value for the display line.
Parameter Type	Float, unit: dBm
Parameter Range	Ref Level ~ Ref Level - 100 dBm

Return	Float, unit: dBm
Default	0 dBm
Menu	Display > Display Line
Example	:DISPlay:WINDow:TRACe:Y:DLINe -10

3.5 Mode Subsection

:INSTrument[:SELect]

Command Format	**:INSTrument[:SELect] SA|MA|DTF|VNA** **:INSTrument[:SELect]?**
Instruction	Sets instrument mode.
Parameter Type	Enumeration
Parameter Range	SA: Spectrum Analyzer MA: Modulation Analyzer DTF: Distance To Fault VNA: Vector Network Analyzer
Return	Enumeration
Default	SA
Menu	mode
Example	:INSTrument DTF

4.Spectrum Analyzer

4.1 Frequency Subsection

[:SENSe]:FREQuency:CENTer

[:SENSe]:FREQuency:STARt

[:SENSe]:FREQuency:STOP

[:SENSe]:FREQuency:CENTer:STEP[:INCRement]

[:SENSe]:FREQuency:CENTer:STEP:AUTO

[:SENSe]:FREQuency:CENTer:SET:STEP

[:SENSe]:FREQuency:OFFSet

[:SENSe]:FREQuency:SPAN

[:SENSe]:FREQuency:SPAN:FULL

[:SENSe]:FREQuency:SPAN:ZERO

[:SENSe]:FREQuency:SPAN:PREVious

[:SENSe]:FREQuency:SPAN:HALF

[:SENSe]:FREQuency:SPAN:DOUBle

Command Format	**[:SENSe]:FREQuency:CENTer <freq>** **[:SENSe]:FREQuency:CENTer?**
Instruction	Sets the center frequency of the spectrum analyzer. Gets the center frequency.

Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	50 Hz~3.199999950 GHz Zero Span: 0 ~ 3.2 GHz
Return	Float, unit: Hz
Default	1.6 GHz
Menu	Frequency > Center Freq
Example	:FREQuency:CENTer 0.2 GHz

Command Format	**[:SENSe]:FREQuency:STARt <freq>** **[:SENSe]:FREQuency:STARt?**
Instruction	Sets the start frequency of the spectrum analyzer. Gets the start Frequency.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	0 Hz ~ 3.199999900 GHz Zero Span: 0 ~ 3.2 GHz
Return	Float, unit: Hz
Default	0 Hz
Menu	Frequency > Start Freq
Example	:FREQuency:STARt 100 Hz

Command Format	**[:SENSe]:FREQuency:STOP <freq>** **[:SENSe]:FREQuency:STOP?**
Instruction	Sets the stop frequency of the spectrum analyzer. Gets the stop frequency.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	100 Hz ~ 3.2 GHz Zero Span: 0 ~ 3.2 GHz
Return	Float, unit: Hz
Default	1.5 GHz
Menu	Frequency > Stop Freq
Example	:FREQuency:STOP 1.0 GHz

Command Format	**[:SENSe]:FREQuency:CENTer:STEP[:INCRement] <freq>** **[:SENSe]:FREQuency:CENTer:STEP[:INCRement]?**
Instruction	Specifies the center frequency step size. Gets the center frequency step.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	1 Hz ~ 3.2 GHz
Return	Float, unit: Hz

Default	320 MHz
Menu	Frequency > Freq Step
Example	:FREQuency:CENTer:STEP 2 MHz

Command Format	**[:SENSe]:FREQuency:CENTer:STEP:AUTO OFF|ON|0|1** **[:SENSe]:FREQuency:CENTer:STEP:AUTO?**
Instruction	Specifies whether the step size is set automatically based on the span. Gets center frequency step mode.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	ON
Menu	Frequency > Freq Step
Example	:FREQuency:CENTer:STEP:AUTO OFF

Command Format	**[:SENSe]:FREQuency:CENTer:SET:STEP**
Instruction	Sets step value equal to center frequency.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Frequency > CF→Step
Example	:FREQuency:CENTer:SET:STEP

Command Format	**[:SENSe]:FREQuency:OFFSet <freq>** **[:SENSe]:FREQuency:OFFSet?**
Instruction	Sets the frequency offset of the spectrum analyzer. Gets the frequency offset.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	-100 GHz ~ 100 GHz
Return	Float, unit: Hz
Default	0 Hz
Menu	Frequency > Freq Offset
Example	:FREQuency:OFFSet 1 GHz

Command Format	**[:SENSe]:FREQuency:SPAN <freq>** **[:SENSe]:FREQuency:SPAN?**
Instruction	Sets the frequency span. Setting the span to 0 Hz puts the analyzer into zero span. Gets span value.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	0 Hz, 100 Hz ~ 3.2GHz
Return	Float, unit: Hz
Default	1.5 GHz
Menu	Span > Span
Example	:FREQuency:SPAN 1 GHz

Command Format	**[:SENSe]:FREQuency:SPAN:FULL**
Instruction	Sets the frequency span to full scale.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Span > Full Span
Example	:FREQuency:SPAN:FULL

Command Format	**[:SENSe]:FREQuency:SPAN:ZERO**
Instruction	Sets the frequency span to zero span.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Span > Zero Span
Example	:FREQuency:SPAN:ZERO

Command Format	**[:SENSe]:FREQuency:SPAN:PREVious**
Instruction	Sets the frequency span to the previous span setting.

Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Span > Last Span
Example	:FREQuency:SPAN:PREVious

Command Format	**[:SENSe]:FREQuency:SPAN:HALF**
Instruction	Sets the frequency span to half of the current span setting.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Span > Zoom In
Example	:FREQuency:SPAN:HALF

Command Format	**[:SENSe]:FREQuency:SPAN:DOUBle**
Instruction	Sets the frequency span to double the current span setting.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Span > Zoom Out
Example	:FREQuency:SPAN:DOUBle

4.2 Amplitude Subsection

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

[:SENSe]:POWer[:RF]:ATTenuation

[:SENSe]:POWer[:RF]:ATTenuation:AUTO

[:SENSe]:POWer[:RF]:GAIN[:STATe]

:DISPlay:WINDow:TRACe:Y:SCALe:RLEVel:OFFSet

:UNIT:POWer

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

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

[:SENSe]:CORRection:OFF

[:SENSe]:CORRection:CSET:ALL[:STATe]

[:SENSe]:CORRection:CSET[1]|2|3|4[:STATe]

[:SENSe]:CORRection:CSET[1]|2|3|4:ADD

[:SENSe]:CORRection:CSET[1]|2|3|4:DELete

[:SENSe]:CORRection:CSET[1]|2|3|4:ALL:DELete

[:SENSe]:CORRection:CSET[1]|2|3|4:DATA

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

Command Format	**:DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel <value>** **:DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel?**
Instruction	This command sets the reference level for the Y-axis. Gets reference level.
Parameter Type	Float, unit: dBm, dBmV, dBuV, dBuA, V, W
Parameter Range	Unit is dBm: -100 dBm ~ 30 dBm Unit is dBmV: -53.01 dBmV ~ 76.99 dBmV Unit is dBuV: 6.99 dBuV ~ 136.99 dBuV Unit is dBuA: -26.99 dBuA ~ 103.01 dBuA Unit is Volts: 2.24 uV ~ 7.07 V Unit is Watts: 100 fW ~ 1 W
Return	Float, unit: dBm
Default	0 dBm
Menu	Amplitude > Ref Level
Example	:DISPlay:WINDow:TRACe:Y:RLEVel 20 DBM

Command Format	**[:SENSe]:POWer[:RF]:ATTenuation <value>** **[:SENSe]:POWer[:RF]:ATTenuation?**
Instruction	Sets the input attenuator of the spectrum analyzer. Gets the input attenuator.
Parameter Type	Integer
Parameter Range	0 dB ~ 51 dB
Return	Integer, unit: dB
Default	20 dB
Menu	Amplitude > Attenuator
Example	:POWer:ATTenuation 10

Command Format	**[:SENSe]:POWer[:RF]:ATTenuation:AUTO OFF|ON|0|1** **[:SENSe]:POWer[:RF]:ATTenuation:AUTO?**
Instruction	This command turns on/off auto input port attenuator state. Gets input port attenuator state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	ON
Menu	Amplitude > Attenuator
Example	:POWer:ATTenuation:AUTO?

Command Format	**[:SENSe]:POWer[:RF]:GAIN[:STATe] OFF|ON|0|1** **[:SENSe]:POWer[:RF]:GAIN[:STATe]?**
Instruction	Turns the internal preamp on/off. Gets preamp on-off state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	OFF
Menu	Amplitude > Preamp
Example	:POWer:GAIN ON

Command Format	**:DISPlay:WINDow:TRACe:Y:SCALe:RLEVel:OFFSet <value>** **:DISPlay:WINDow:TRACe:Y:SCALe:RLEVel:OFFSet?**
Instruction	Sets reference offsets. Gets reference offsets.
Parameter Type	Float
Parameter Range	-100dB~100dB
Return	Float, unit: dB
Default	0dB
Menu	Amplitude > Ref OffSets
Example	:DISPlay:WINDow:TRACe:Y:SCALe:RLEVel:OFFSet 2

Command Format	**:UNIT:POWer DBM|DBMV|DBUV|V|W** **:UNIT:POWer?**
Instruction	Specifies amplitude units for the input, output and display. Gets amplitude units.

Parameter Type	Enumeration
Parameter Range	DBM|DBMV|DBUV|DBUA|V|W,
Return	Enumeration
Default	DBM
Menu	Amplitude > Units
Example	:UNIT:POWer DBMV

Command Format	**:DISPlay:WINDow:TRACe:Y[:SCALe]:SPACing LINear|LOGarithmic** **:DISPlay:WINDow:TRACe:Y[:SCALe]:SPACing?**
Instruction	Toggles the vertical graticule divisions between logarithmic unit and linear unit. The default logarithmic unit is dBm, and the linear unit is V. Gets scale type.
Parameter Type	Enumeration
Parameter Range	LINear|LOGarithmic
Return	Enumeration
Default	LOGarithmic
Menu	Amplitude > Scale Type
Example	:DISPlay:WINDow:TRACe:Y:SPACing LINear

Command Format	**:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision <integer>** **:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision?**
Instruction	This command sets the per-division display scaling for the y-axis when scale type of Y axis is set to Log. Gets Scale/Div when scale type of Y axis is set to Log.
Parameter Type	Float
Parameter Range	1 dB ~ 10 dB
Return	Float, unit: dB
Default	10 dB
Menu	Amplitude > Scale/Div
Example	:DISPlay:WINDow:TRACe:Y:PDIVision 10 dB

Command Format	**[:SENSe]:CORRection:OFF**
Instruction	Turns off the amplitude correction function off and all of the correction sets are off.
Parameter Type	None
Parameter Range	None

Return	None
Default	None
Menu	None
Example	:SENSe:CORRection:OFF

Command Format	**[:SENSe]:CORRection:CSET:ALL[:STATe] OFF|ON|0|1** **[:SENSe]:CORRection:CSET:ALL[:STATe]?**
Instruction	Turns on or off the amplitude corrections. When turned on, only the correction sets that were turned on are enabled. When turned off, all of the correction Sets are disabled. If there is no correction enabled, state cannot be set to on.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	OFF
Menu	Amplitude > Corrections > Apply Corrections
Example	:SENSe:CORRection:CSET:ALL:STATe OFF

Command Format	**[:SENSe]:CORRection:CSET[1]|2|3|4[:STATe]** **[:SENSe]:CORRection:CSET[1]|2|3|4[:STATe]?**
Instruction	Turns the amplitude correction function on/off. Gets the amplitude correction function state.
Parameter Type	None
Parameter Range	None
Return	0|1
Default	OFF
Menu	Amplitude > Corrections > Correction1|2|3|4
Example	:CORRection:CSET2:OFF

Command Format	**[:SENSe]:CORRection:CSET[1]|2|3|4:ADD <x1,y1,x2,y2;…>**
Instruction	Adds Correction Points.
Parameter Type	String<freq, ampl,freq, ampl,freq, ampl,⋯⋯>
Parameter Range	None
Return	
Default	None

Menu	Amplitude > Corrections > CorrectionX > Add Point
Example	:CORRection:CSET2:ADD 10000000,-10,15000000,-12

Command Format	**[:SENSe]:CORRection:CSET[1]|2|3|4:DELete <index>**
Instruction	Deletes Correction Points.
Parameter Type	Serial number of Correction Points
Parameter Range	None
Return	
Default	None
Menu	Amplitude > Corrections > CorrectionX > Del Point
Example	:CORRection:CSET2: DELete 2

Command Format	**[:SENSe]:CORRection:CSET[1]|2|3|4:ALL:DELete**
Instruction	Deletes All Correction Points.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Amplitude > Corrections > CorrectionX > Del All
Example	:CORRection:CSET2: ALL:DELete

Command Format	**[:SENSe]:CORRection:CSET[1]|2|3|4:DATA <x1,y1,x2,y2;…>** **[:SENSe]:CORRection:CSET[1]|2|3|4:DATA?**
Instruction	Sets correction X data. Reads correction X data.
Parameter Type	None
Parameter Range	None
Return	String
Default	None
Menu	None
Example	:CORRection:CSET2:DATA?

Command Format	**[:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude] OHM50 |OHM75** **[:SENSe]:CORRection:IMPedance[:INPut][:MAGNitude]?**
Instruction	Sets the input impedance for voltage-to-power conversions. Gets the input impedance.
Parameter Type	Enumeration
Parameter Range	OHM50|OHM75
Return	OHM50|OHM75
Default	OHM50
Menu	Amplitude > Corrections
Example	CORRection:IMPedance?

4.3 Sweep Subsection

[:SENSe]:SWEep:MODE

[:SENSe]:SWEep:TIME

[:SENSe]:SWEep:TIME:AUTO

[:SENSe]:SWEep:SPEed

[:SENSe]:SWEep:COUNt

[:SENSe]:QPD:DWELl:TIME

:INITiate[:IMMediate]

:INITiate:RESTart

:INITiate:CONTinuous

:INITiate:Pause

:INITiate:RESume

ABORt

Command Format	**[:SENSe]:SWEep:MODE AUTO|FFT|SWEep** **[:SENSe]:SWEep:MODE?**
Instruction	Sets sweep mode. Gets sweep mode.
Parameter Type	Enumeration
Parameter Range	AUTO|FFT|SWEep
Return	Enumeration
Default	SWEep
Menu	Sweep
Example	:SWEep:MODE SWEep

Command Format	**[:SENSe]:SWEep:TIME <time>** **[:SENSe]:SWEep:TIME?**
Instruction	Specifies the time in which the instrument sweeps the display. A span value of 0 Hz causes the analyzer to enter zero span mode. In zero span the X-axis represents time rather than frequency.
Parameter Type	Float, unit: ks, s, ms, us
Parameter Range	450us ~ 1500 s
Return	Float, unit: s
Default	312.416ms(216.288ms, 192.256ms, 168.224ms, 120.160ms)
Menu	Sweep > Sweep Time
Example	:SWEep:TIME 5s

Command Format	**[:SENSe]:SWEep:TIME:AUTO OFF|ON|0|1** **[:SENSe]:SWEep:TIME:AUTO?**
Instruction	This command turns on/off auto sweep time state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	ON
Menu	Sweep > Sweep Time
Example	:SWEep:TIME:AUTO ON

Command Format	**[:SENSe]:SWEep:SPEed NORMal|ACCUracy** **[:SENSe]:SWEep:SPEed?**
Instruction	Toggles the sweep speed between normal and accuracy.
Parameter Type	Enumeration
Parameter Range	ACCUracy|NORMal
Return	Enumeration
Default	NORMal
Menu	Sweep > Sweep Rule
Example	:SWEep: SPEed NORMal

Command Format	**[:SENSe]:SWEep:COUNt <integer>** **[:SENSe]:SWEep:COUNt?**
Instruction	Sets sweep numbers, when single sweep on. Gets sweep numbers, when single sweep on.

Parameter Type	Integer
Parameter Range	1 ~ 99999
Return	Integer
Default	1
Menu	Sweep > Numbers
Example	:SWEep:COUNt 10

Command Format	**[:SENSe]:QPD:DWELl:TIME < time >** **[:SENSe]:QPD:DWELl:TIME?**
Instruction	Sets QPD Time. Gets QPD Time.
Parameter Type	Float, unit: s, ms, us
Parameter Range	0 us ~ 10 s(qusai-peak: 900 us ~ 30 ks)
Return	Float, unit: s
Default	50 ms
Menu	Sweep > QPD Time
Example	:QPD:DWELl:TIME 10s

Command Format	**:INITiate[:IMMediate]**
Instruction	Restarts the current sweep.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	
Example	:INITiate:IMMediate

Command Format	**:INITiate:RESTart**
Instruction	Restarts the current sweep. :INITiate:RESTart and :INITiate:IMMediate perform exactly the same function.
Parameter Type	None
Parameter Range	None
Return	None

Default	None
Menu	
Example	:INITiate:RESTart

Command Format	**:INITiate:CONTinuous OFF|ON|0|1** **:INITiate:CONTinuous?**
Instruction	Sets continuous sweep mode on-off. Gets continuous sweep mode state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	ON
Menu	Sweep > Sweep
Example	:INITiate:CONTinuous OFF

Command Format	**:INITiate:Pause**
Instruction	Pause current sweeeep(pause at the end of the current sweep).
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	
Example	:INITiate:Pause

Command Format	**:INITiate:RESume**
Instruction	Resumes paused sweep.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	
Example	:INITiate:RES

Command Format	**ABORt**
Instruction	This command is used to stop the current measurement. It aborts the currentmeasurement as quickly as possible, resets the sweep and trigger systems, and puts the measurement into an "idle" state. If the analyzer is set for Continuous measurement, it sets up the measurement and initiates a new data measurement sequence with a new data acquisition (sweep) taken once the trigger condition is met. If the analyzer is set for Single measurement, it remains in the "idle" state until an :INIT:IMM command is received.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	None
Default	ABORt

4.4 Trigger Subsystem

:TRIGger[:SEQuence]:SOURce

:TRIGger[:SEQuence]:VIDeo:LEVel

:TRIGger[:SEQuence]:RFBurst:SLOPe

Command Format	**:TRIGger[:SEQuence]:SOURce IMMediate|VIDeo|EXTernal** **:TRIGger[:SEQuence]:SOURce?**
Instruction	Specifies the source (or type) of triggering used to start a measurement. Gets trigger type.
Parameter Type	Enumeration
Parameter Range	IMMediate: free-run triggering. VIDeo: triggers on the video signal level. EXTernal: allows you to connect an external trigger source.
Return	Enumeration
Default	IMMediate
Menu	Trigger
Example	:TRIGger:SOURce IMMediate

Command Format	**:TRIGger[:SEQuence]:VIDeo:LEVel <value>** **:TRIGger[:SEQuence]:VIDeo:LEVel?**
Instruction	Specifies the level at which a video trigger will occur. Video is adjusted using this command, but must also be selected using the command.

	Gets video Trigger Level.
Parameter Type	Float, unit: dBm, dBmV, dBuV, dBuA, V, W
Parameter Range	Unit is dBm: -300 dBm ~ 50 dBm Unit is dBmV: -253.01 dBmV ~ 96.99 dBmV Unit is dBuV: -193.01 dBuV ~ 156.99 dBuV Unit is dBuA: -226.99 dBuA ~ 123.01 dBuA Unit is Volts: 223E-16V ~ 70.71 V Unit is Watts: 1.00E-33 W ~ 100 W
Return	Float, unit: dBm
Default	0 dBm
Menu	Trigger > Video Level
Example	:TRIGger:VIDeo:LEVel 0.5 dBm

Command Format	**:TRIGger[:SEQuence]:RFBurst:SLOPe POSitive|NEGative** **:TRIGger[:SEQuence]:RFBurst:SLOPe?**
Instruction	This command activates the trigger condition that allows the next sweep to start when the external voltage (connected to EXT TRIG IN connector) passes through approximately 1.5 volts. The external trigger signal must be a 0V to +5V TTL signal. This function only controls the trigger polarity (for positive or negative-going signals). Gets Trigger edge.
Parameter Type	Enumeration
Parameter Range	POSitive: positive edge. NEGative: negative edge.
Return	Enumeration
Default	POSitive
Menu	Trigger > External Trigger
Example	:TRIGger:RFBurst:SLOPe POSitive

4.5 Bandwidth Subsection

[:SENSe]:BWIDth[:RESolution]

[:SENSe]:BWIDth[:RESolution]:AUTO

[:SENSe]:BWIDth:VIDeo

[:SENSe]:BWIDth:VIDeo:AUTO

[:SENSe]:BWIDth:VIDeo:RATio

[:SENSe]:BWIDth:VIDeo:RATio:CONfig?

[:SENSe]:FILTer:TYPE

Command Format	**[:SENSe]:BWIDth[:RESolution] <freq>** **[:SENSe]:BWIDth[:RESolution]?**
Instruction	Specifies the resolution bandwidth. For numeric entries, all RBW types choose the nearest (arithmetically, on a linear scale, rounding up) available RBW to the value entered.

	Gets the resolution bandwidth.
Parameter Type	Discrete
Parameter Range	1Hz, 3Hz, 10 Hz, 30 Hz, 100 Hz, 300 Hz, 1 kHz, 3 kHz, 10 kHz, 30 kHz, 100 kHz, 300 kHz, 1 MHz
Return	Float, unit: Hz
Default	1 MHz
Menu	BW > RBW
Example	:BWIDth 1 kHz

Command Format	**[:SENSe]:BWIDth[:RESolution]:AUTO OFF|ON|0|1** **[:SENSe]:BWIDth[:RESolution]:AUTO?**
Instruction	Turns on/off auto resolution bandwidth state. Gets the resolution bandwidth state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	ON
Menu	BW > RBW
Example	:BWID:AUTO On

Command Format	**[:SENSe]:BWIDth:VIDeo <freq>** **[:SENSe]:BWIDth:VIDeo?**
Instruction	Specifies the video bandwidth. Gets the video bandwidth.
Parameter Type	Discrete
Parameter Range	1 Hz, 3 Hz, 10 Hz, 30 Hz, 100 Hz, 300 Hz, 1 kHz, 3 kHz, 10 kHz, 30 kHz, 100 kHz, 300 kHz, 1 MHz
Return	Float, unit: Hz
Default	1 MHz
Menu	BW > VBW
Example	:BWIDth:VIDeo 10 KHZ

Command Format	**[:SENSe]:BWIDth:VIDeo:AUTO OFF|ON|0|1** **[:SENSe]:BWIDth:VIDeo:AUTO?**
Instruction	This command turns on/off auto video bandwidth state. Gets the video bandwidth state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1

Return	0|1
Default	ON
Menu	BW > VBW
Example	BWIDth:VIDeo:AUTO OFF

Command Format	**[:SENSe]:BWIDth:VIDeo:RATio <number> [:SENSe]:BWIDth:VIDeo:RATio?**
Instruction	Specifies the ratio of the video bandwidth to the resolution bandwidth. Gets the ratio of the video bandwidth to the resolution bandwidth.
Parameter Type	Discrete, Float
Parameter Range	0.001, 0.003, 0.01, 0.03, 0.1, 0.3, 1.0, 3.0, 10.0, 30.0, 100.0, 300.0, 1000.0
Return	Float
Default	1.0
Menu	BW > VBW/RBW
Example	:BWIDth:VIDeo:RATio 30

Command Format	**[:SENSe]:BWIDth:VIDeo:RATio:CONfig?**
Instruction	This command turns on/off auto video to resolution bandwidth ratio.
Parameter Type	None
Parameter Range	None
Return	0|1
Default	1
Menu	None
Example	:BWIDth:VIDeo:RATio:CONfig?

Command Format	**[:SENSe]:FILTer:TYPE EMI|GAUSS [:SENSe]:FILTer:TYPE?**
Instruction	Sets filter type Gets filter type
Parameter Type	Enumeration
Parameter Range	EMI|GAUSS
Return	Enumeration
Default	GAUSS
Menu	BW > Filter

Example	:FILTer:TYPE EMI

4.6 Trace Subsection

:TRACe[1]|2|3|4:MODE

:TRACe[:DATA]?

:FORMat[:TRACe][:DATA]

:CALCulate[:SELected]:MATH:FUNCtion

:TRACe:MATH:X

:TRACe:MATH:Y

:TRACe:MATH:Z

:TRACe:MATH:OFFSet

[:SENSe]:FREQuency:TUNE:IMMediate

[:SENSe]:DETector:TRACe[1]|2|3|4[:FUNCtion]

[:SENSe]:AVERage:TYPE

[:SENSe]:AVERage:TRACe[1]|2|3|4:COUNt

[:SENSe]:AVERage:TRACe[1]|2|3|4?

[:SENSe]:AVERage:TRACe[1]|2|3|4:CLEar

Command Format	**:TRACe[1]|2|3|4:MODE WRITe|MAXHold|MINHold|VIEW|BLANk|AVERage** **:TRACe[1]|2|3|4:MODE?**
Instruction	Selects the display mode for the selected trace. Gets the display mode of the seletcted trace.
Parameter Type	Enumeration
Parameter Range	WRITe: puts the trace in the normal mode, updating the data. MAXHold: displays the highest measured trace value for all the data that has been measured since the function was turned on. MINHold: displays the lowest measured trace value for all the data that has been measured since the function was turned on. VIEW: turns on the trace data so that it can be viewed on the display. BLANk: turns off the trace data so that it is not viewed on the display. AVERage: averages the trace for test period.
Return	Enumeration
Default	Trace1:WRITe, Trace2|3|4: BLANk
Menu	Trace
Example	:TRAC1:MODE VIEW

Command Format	**:TRACe[:DATA]? 1|2|3|4**
Instruction	This query command returns the current displayed data. You can also add trace parameters directly after trace as

	:TRACe[1]|2|3|4[:DATA]?
Parameter Type	Enumeration
Parameter Range	1|2|3|4 or A|B|C|D or TRACE1| TRACE2| TRACE3| TRACE4
Return	String
Default	1
Menu	None
Example	:TRACe:DATA? 1

Command Format	**:FORMat[:TRACe][:DATA] ASCii|REAL** **:FORMat[:TRACe][:DATA]?**
Instruction	Sets trace data type. Gets trace data type.
Parameter Type	Enumeration
Parameter Range	ASCii REAL: single precision floating-point (float)
Return	String
Default	ASCii
Menu	None
Example	:FORMat ASCii

Command Format	**:CALCulate[:SELected]:MATH:FUNCtion** **:CALCulate[:SELected]:MATH:FUNCtion?**
Instruction	Sets trace math function. Gets trace data function.
Parameter Type	Enumeration
Parameter Range	OFF: Trace Math Off PDIF : Power Diff PSUM : Power Sum LOFF : Log Offset LDIF : Log Diff
Return	Enumeration
Default	OFF
Menu	Trace > Math
Example	:CALCulate[:SELected]:MATH:FUNCtion?

Command Format	**:TRACe:MATH:X A|B|C|D** **:TRACe:MATH:X?**
Instruction	Sets trace math input X. Gets trace math input X.
Parameter	Enumeration

Type	
Parameter Range	A|B|C|D or TRACE1|TRACE2|TRACE3|TRACE4
Return	Enumeration
Default	A
Menu	Trace > Math > Input X
Example	:TRACe:MATH:X A

Command Format	**:TRACe:MATH:Y A|B|C|D** **:TRACe:MATH:Y?**
Instruction	Sets trace math input Y. Gets trace math input Y.
Parameter Type	Enumeration
Parameter Range	A|B|C|D or TRACE1|TRACE2|TRACE3|TRACE4
Return	Enumeration
Default	A
Menu	Trace > Math > Input Y
Example	:TRACe:MATH:Y A

Command Format	**:TRACe:MATH:Z A|B|C|D** **:TRACe:MATH:Z?**
Instruction	Sets trace math Output Z. Gets trace math Output Z.
Parameter Type	Enumeration
Parameter Range	A|B|C|D or TRACE1|TRACE2|TRACE3|TRACE4
Return	Enumeration
Default	A
Menu	Trace > Math > Output Z
Example	:TRACe:MATH:Z A

Command Format	**:TRACe:MATH:OFFSet <const>** **:TRACe:MATH:OFFSet?**
Instruction	Sets trace math OFFSet. Gets trace math OFFSet.
Parameter Type	Float
Parameter Range	-100 dB ~100 dB
Return	Float

Default	0.00 dB
Menu	Trace > Math
Example	:TRACe:MATH:OFFSet 7

Command Format	**[:SENSe]:FREQuency:TUNE:IMMediate**
Instruction	Auto tune the spectrum analyzer parameter to display the main signal.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Auto Tune
Example	:FREQuency:TUNE:IMMediate

Command Format	**[:SENSe]:DETector:TRACe[1]|2|3|4[:FUNCtion] NEGative|POSitive|SAMPle|AVERage|NORMAL|QUASi** **[:SENSe]:DETector:TRACe[1]|2|3|4[:FUNCtion]?**
Instruction	Specifies the detection mode. For each trace interval (bucket), average detection displays the average of all the samples within the interval. Gets the detection mode.
Parameter Type	Enumeration
Parameter Range	NEGative: Negative peak detection displays the lowest sample taken during the interval being displayed. POSitive: Positive peak detection displays the highest sample taken during the interval being displayed. SAMPle: Sample detection displays the sample taken during the interval being displayed, and is used primarily to display noise or noise-like signals. In sample mode, the instantaneous signal value at the present display point is placed into memory. This detection should not be used to make the most accurate amplitude measurement of non noise-like signals. AVERage: Average detection is used when measuring the average value of the amplitude across each trace interval (bucket). The averaging method used by the average detector is set to either video or power as appropriate when the average type is auto coupled. NORMAL: Normal detection selects the maximum and minimum video signal values alternately. When selecting Normal detection,”Norm”appears in the upper-left corner. QUASi: Quasipeak detection is a form of detection where a signal level is weighted based on the repetition frequency of the spectral components making up the signal. That is to say, the result of a quasi-peak measurement depends on the repetition rate of the signal.
Return	Enumeration
Default	POSitive
Menu	Detect
Example	:DETector:TRACe QUAS

Command Format	**[:SENSe]:AVERage:TYPE LOGPower|POWer|VOLTage** **[:SENSe]:AVERage:TYPE?**
Instruction	Toggles the average type between Log power, power and voltage.
Parameter Type	Enumeration
Parameter Range	LOGPower|POWer|VOLTage
Return	Enumeration
Default	LOGPower
Menu	BW > Avg Type
Example	AVERage:TYPE VOLTage

Command Format	**[:SENSe]:AVERage:TRACe[1]|2|3|4:COUNt <integer>** **[:SENSe]:AVERage:TRACe[1]|2|3|4:COUNt?**
Instruction	Specifies the number of measurements that are combined. Gets the number of measurements that are combined.
Parameter Type	Integer
Parameter Range	1 ~ 999
Return	Integer
Default	1
Menu	Trace > Average
Example	:AVERage:TRACe1:COUNt 10

Command Format	**[:SENSe]:AVERage:TRACe[1]|2|3|4?**
Instruction	Get the current average number of traces.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	None
Example	:AVERage:TRACe?

Command Format	**[:SENSe]:AVERage:TRACe[1]|2|3|4:CLEar**
Instruction	Restarts the trace average. This command is only available when average is on.

Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	None
Example	:AVERage:TRAC1:CLEar

4.7 Marker Subsection

:CALCulate:MARKer[1]|2|3|4|5|6|7|8:STATe

:CALCulate:MARKer:AOFF

:CALCulate:MARKer[1]|2|3|4|5|6|7|8:MODE

:CALCulate:MARKer[1]|2|3|4|5|6|7|8:TRACe

:CALCulate:MARKer[1]|2|3|4|5|6|7|8:RELative:TO:MARKer

:CALCulate:MARKer[1]|2|3|4|5|6|7|8:X

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

:CALCulate:MARKer:TABLe

:CALCulate:MARKer[1]|2|3|4|5|6|7|8[:SET]:START

:CALCulate:MARKer[1]|2|3|4|5|6|7|8[:SET]:STOP

:CALCulate:MARKer[1]|2|3|4|5|6|7|8[:SET]:CENTer

:CALCulate:MARKer[1]|2|3|4|5|6|7|8[:SET]:STEP

:CALCulate:MARKer[1]|2|3|4|5|6|7|8[:SET]:RLEVel

:CALCulate:MARKer[1]|2|3|4|5|6|7|8:DELTa[:SET]:SPAN

:CALCulate:MARKer[1]|2|3|4|5|6|7|8:DELTa[:SET]:CENTer

:CALCulate:MARKer:PEAK:SEARch:MODE

:CALCulate:MARKer:PEAK:SORT

:CALCulate:MARKer:PEAK:THReshold

:CALCulate:MARKer:PEAK:EXCursion

:CALCulate:MARKer:PEAK:TABLe

:CALCulate:PEAK:TABLe?

:CALCulate:MARKer[1]|2|3|4|5|6|7|8:CPEak[:STATe]

:CALCulate:MARKer[1]|2|3|4|5|6|7|8:MAXimum

:CALCulate:MARKer[1]|2|3|4|5|6|7|8:MAXimum:NEXT

:CALCulate:MARKer[1]|2|3|4|5|6|7|8:MAXimum:LEFT

:CALCulate:MARKer[1]|2|3|4|5|6|7|8:MAXimum:RIGHt

:CALCulate:MARKer[1]|2|3|4|5|6|7|8:CPSearch

:CALCulate:MARKer[1]|2|3|4|5|6|7|8:FUNCtion

:CALCulate:MARKer:FCOunt[:STATe]

:CALCulate:MARKer:FCOunt:X?

:CALCulate:MARKer[1]|2|3|4|5|6|7|8:BANDwidth:RESult?

:CALCulate:MARKer[1]|2|3|4|5|6|7|8:BANDwidth:NDB

:CALCulate:MARKer[1]|2|3|4|5|6|7|8:X:READout

:CALCulate:MARKer:TRCKing[:STATe]

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8:STATe OFF|ON|0|1** **:CALCulate:MARKer[1]|2|3|4|5|6|7|8:STATe?**
Instruction	This command toggles the selected marker state between on and off. Gets marker state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	OFF
Menu	Marker
Example	:CALCulate:MARK1:STATe ON

Command Format	**:CALCulate:MARKer:AOFF**
Instruction	Turns all the markers off.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	None
Example	:CALCulate:MARKer:AOFF

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8:MODE POSition|DELTa|FIXed|OFF** **:CALCulate:MARKer[1]|2|3|4|5|6|7|8:MODE?**
Instruction	Selects the type of the selected marker that you want to activate. Gets the type of the selected marker.
Parameter Type	Enumeration
Parameter Range	POSition: selects a normal marker that can be positioned on a trace and from which trace information will be generated. DELTa: activates a pair of markers, one of which is fixed at the current marker location.

	The other marker can then be moved around on the trace. The marker readout shows the marker value which moves. FIXed: Active marker fixed at current position. OFF: turns the designated marker off. If a marker is not active when the mode is queried, “off” will be returned.
Return	Enumeration
Default	OFF
Menu	Marker
Example	:CALCulate:MARK1:MODE POSition

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8:TRACe 1|2|3|4** **:CALCulate:MARKer[1]|2|3|4|5|6|7|8:TRACe?**
Instruction	This command assigns the specified marker to the designated trace 1, 2, 3 or 4. Gets the specified marker to which trace.
Parameter Type	Enumeration
Parameter Range	MARKer:1|2|3|4|5|6|7|8 TRACe:1|2|3|4
Return	Enumeration
Default	1
Menu	Marker > Select Trace
Example	CALCulate:MARK1:TRAC 1

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8:RELative:TO:MARKer 1|2|3|4|5|6|7|8** **:CALCulate:MARKer[1]|2|3|4|5|6|7|8:RELative:TO:MARKer?**
Instruction	Sets marker relative to. Gets marker relative to.
Parameter Type	Enumeration
Parameter Range	1|2|3|4|5|6|7|8
Return	Enumeration
Default	1
Menu	Marker > Relative To
Example	:CALCulate:MARKer1:RELative:TO:MARK 3

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8:X <para>** **:CALCulate:MARKer[1]|2|3|4|5|6|7|8:X?**
Instruction	This command positions the designated marker on its assigned trace at the specified trace X value. The value is in the X-axis units, which can be a frequency or time. The query returns the current X value of the designated marker. When the readout mode is frequency, the query returns the X value of the span of the marker in integer and the unit is “Hz”. When the readout mode is time or period, the query returns the X value of the span of the

	marker in scientific notation and the unit is “s”. Reference Command: :CALCulate:MARKer[1]|2|3|4|5|6|7|8:X:READout
Parameter Type	Frequency: Float, unit: Hz, kHz, MHz, GHz, Default “Hz” Time: Float, unit: us, ms, s, ks, Default “s”
Parameter Range	0 Hz ~ 1.5 GHz or 10 ms ~ 1000 s
Return	Float
Default	750 MHz or 312.64 ms
Menu	Marker > Normal
Example	:CALCulate:MARKer4:X 0.4 GHz :CALCulate:MARKer4:X 200 ms :CALCulate:MARKer4:X?

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8:Y?**
Instruction	This command reads the current Y value for the designated marker. This command can be used to read the results of noise marker. Make sure that Marker is on, Reference Command: :CALCulate:MARKer[1]|2|3|4|5|6|7|8:STATe :CALCulate:MARKer[1]|2|3|4|5|6|7|8:MODE
Parameter Type	None
Parameter Range	None
Return	Float, unit: dBm
Default	None
Menu	None
Example	:CALCulate:MARKer1:Y? Return: -25

Command Format	**:CALCulate:MARKer:TABLe ON|OFF|0|1** **:CALCulate:MARKer:TABLe?**
Instruction	Toggles the marker table between on and off. Gets the status of the marker table.
Parameter Type	Boolean
Parameter Range	ON|OFF|0|1
Return	0|1
Default	0
Menu	Marker > Marker Table
Example	:CALCulate:MARKer:TABLe ON

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8[:SET]:START**

Instruction	Sets the start frequency to the value of the specified marker frequency. This command is not available in zero span. If the specified Marker is OFF, it will set the marker on center.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Marker→ > M→Start Freq
Example	:CALCulate:MARKer1:START

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8[:SET]:STOP**
Instruction	Sets the stop frequency to the value of the specified marker frequency. This command is not available in zero span . If the specified Marker is OFF, it will set the marker on center.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Marker→ > M→Stop Freq
Example	:CALCulate:MARKer1:STOP

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8[:SET]:CENTer**
Instruction	This command sets the center frequency equal to the specified marker frequency, which moves the marker to the center of the screen. This command is not available in zero span. If the specified Marker is OFF, it will set the marker on center.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Marker→ > M→CF
Example	:CALCulate:MARKer1:CENTer

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8[:SET]:STEP**
Instruction	This command sets the center frequency step equal to the specified marker frequency. This command is not available in zero span. If the specified Marker is OFF, it will set the marker on center.

Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Marker→ > M→CF Step
Example	:CALCulate:MARKer1:STEP

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8[:SET]:RLEVel**
Instruction	This command sets the reference level equal to the specified marker frequency. If the specified Marker is OFF, it will set the marker on center.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Marker→ > M→Ref Level
Example	:CALCulate:MARKer2:RLEVel

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8:DELTa[:SET]:SPAN**
Instruction	This command sets the span equal to the specified delta marker frequency. This command can be only used in DELTa marker mode, Reference Command:CALCulate:MARKer[1]|2|3|4|5|6|7|8:MODE
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Marker→ > △ M→Span
Example	:CALCulate:MARKer2:DELTa:SPAN

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8:DELTa[:SET]:CENTer**
Instruction	This command sets the center frequency equal to the specified delta marker frequency. This command can be only used in DELTa marker mode, Reference Command:CALCulate:MARKer[1]|2|3|4|5|6|7|8:MODE
Parameter Type	None
Parameter Range	None

Return	None
Default	None
Menu	Marker→ > △ M→CF
Example	:CALCulate:MARKer3:DELTa:CENTer

Command Format	**:CALCulate:MARKer:PEAK:SEARch:MODE MAXimum|MINimum** **:CALCulate:MARKer:PEAK:SEARch:MODE?**
Instruction	This is for the analyzer's internal peak identification routine to recognize a signal as a peak.
Parameter Type	Enumeration
Parameter Range	MAXimum|MINimum
Return	Enumeration
Default	MAXimum
Menu	Peak > Search Config > Peak Type
Example	:CALCulate:MARKer:PEAK:SEARch:MODE MINimum

Command Format	**:CALCulate:MARKer:PEAK:SORT FREQuency |AMPLitude** **:CALCulate:MARKer:PEAK:SORT?**
Instruction	Sets the type of peak sort by. Gets the type of peak sort by.
Parameter Type	Enumeration
Parameter Range	FREQuency: Frequency AMPLitude: Amplitude
Return	Enumeration: FREQ|AMPL
Default	AMPL
Menu	Peak > Search Config > Sort By
Example	:CALCulate:MARKer:PEAK:SORT FREQ

Command Format	**:CALCulate:MARKer:PEAK:THReshold <value>** **:CALCulate:MARKer:PEAK:THReshold?**
Instruction	Specifies the minimum signal level for the analyzers internal peak identification routine to recognize a signal as a peak. This applies to all traces and all windows. Gets the minimum signal level for the analyzers internal peak identification routine to recognize a signal as a peak.
Parameter Type	Float, unit: dBm
Parameter Range	-200.0 dBm~ 200.0 dBm
Return	Float, unit: dBm
Default	-160.0 dBm

Menu	Peak > Search Config > Peak Threshold
Example	:CALCulate:MARKer:PEAK:THReshold -50

Command Format	**:CALCulate:MARKer:PEAK:EXCursion <value>** **:CALCulate:MARKer:PEAK:EXCursion?**
Instruction	Specifies the minimum signal excursion above the threshold for the internal peak identification routine to recognize a signal as a peak.
Parameter Type	Float, unit: dB
Parameter Range	0 ~ 200.0 dB
Return	Float, unit: dB
Default	0 dB
Menu	Peak > Search Config > Peak Excursion
Example	:CALCulate:MARKer:PEAK:EXCursion 10

Command Format	**:CALCulate:MARKer:PEAK:TABLe ON|OFF|0|1** **:CALCulate:MARKer:PEAK:TABLe?**
Instruction	Toggles the peak table between on and off. Gets the status of the peak table.
Parameter Type	Boolean
Parameter Range	ON|OFF|0|1
Return	0|1
Default	0
Menu	Peak > Peak Table

Command Format	**:CALCulate:PEAK:TABLe?**
Instruction	Gets peak table data.
Parameter Type	None
Parameter Range	None
Return	String
Default	None
Menu	Peak > Peak Table
Example	:CALCulate:PEAK:TABLe?

Command	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8:CPEak[:STATe] OFF|ON|0|1**

Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8:CPEak[:STATe]?**
Instruction	Toggles the continuous peak search function between on and off. Gets the continuous peak search function state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	None
Menu	Peak > Cont Peak
Example	:CALCulate:MARKer1:CPEak ON

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8:MAXimum**
Instruction	Performs a peak search based on the search mode settings. (based on the search mode settings, include: peak search mode, peak threshold and peak excursion, Reference Commands: :CALCulate:MARKer:PEAK:SEARch:MODE :CALCulate:MARKer:PEAK:THReshold :CALCulate:MARKer:PEAK:EXCursion)
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Peak
Example	:CALCulate:MARKer4:MAXimum

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8:MAXimum:NEXT**
Instruction	Places the selected marker on the next highest signal peak of the current marked peak. (based on the search mode settings, include: peak search mode, peak threshold and peak excursion, Reference Commands: :CALCulate:MARKer:PEAK:SEARch:MODE :CALCulate:MARKer:PEAK:THReshold :CALCulate:MARKer:PEAK:EXCursion)
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Peak > Next Peak
Example	:CALCulate:MARKer1:MAXimum:NEXT

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8:MAXimum:LEFT**
Instruction	Places the selected marker on the next highest signal peak to the left of the current marked peak. (based on the search mode settings, include: peak search mode, peak threshold and peak excursion, Reference Commands: :CALCulate:MARKer:PEAK:SEARch:MODE :CALCulate:MARKer:PEAK:THReshold :CALCulate:MARKer:PEAK:EXCursion)
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Peak > Left Peak
Example	:CALCulate:MARKer1:MAXimum:LEFT

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8:MAXimum:RIGHt**
Instruction	Places the selected marker on the next highest signal peak to the right of the current marked peak. (based on the search mode settings, include: peak search mode, peak threshold and peak excursion, Reference Commands: :CALCulate:MARKer:PEAK:SEARch:MODE :CALCulate:MARKer:PEAK:THReshold :CALCulate:MARKer:PEAK:EXCursion)
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Peak > Right Peak
Example	:CALCulate:MARKer1:MAXimum:RIGHt

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8:CPSearch**
Instruction	Positions a pair of delta markers on the highest and lowest points on the trace.
Parameter Type	None
Parameter Range	None
Return	None
Default	None

Menu	Peak > Peak Peak
Example	:CALCulate:MARKer1:CPSearch

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8:FUNCtion OFF|FCOunt|NOISe|NDB** **:CALCulate:MARKer[1]|2|3|4|5|6|7|8:FUNCtion?**
Instruction	This command selects the marker function for the designated marker. Gets the selected marker function for the designated marker.
Parameter Type	Enumeration
Parameter Range	OFF: refers to the normal function. FCOunt: refers to the frequency counter function. NOISe: refers to the noise measurement function. NDB: refers to the N dB bandwidth function.
Return	Enumeration
Default	OFF
Menu	Marker Fn
Example	:CALCulate:MARK1:FUNCtion FCOunt

Command Format	**:CALCulate:MARKer:FCOunt[:STATe] ON|OFF|0|1**
Instruction	To set the frequency counter status.
Parameter Type	Boolean
Parameter Range	ON|OFF|0|1
Return	0|1
Default	0
Menu	Marker Fn > Freq Counter
Example	:CALCulate:MARK:FCOunt 1

Command Format	**:CALCulate:MARKer:FCOunt:X?**
Instruction	To query the frequency counter.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Marker Fn > Freq Counter
Example	:CALCulate:MARK:FCOunt:X?

Command Format	:CALCulate:MARKer[1]|2|3|4|5|6|7|8:BANDwidth:RESult?
Instruction	Gets the result of N dB bandwidth measurement.
Parameter Type	None
Parameter Range	None
Return	Float
Default	None
Menu	Marker Fn > N dB BW
Example	:CALCulate:MARK1:BANDwidth:RESult?

Command Format	:CALCulate:MARKer[1]|2|3|4|5|6|7|8:BANDwidth:NDB <value> :CALCulate:MARKer[1]|2|3|4|5|6|7|8:BANDwidth:NDB?
Instruction	Sets the reference value of N dB bandwidth measurement. Gets the reference value of N dB bandwidth measurement.
Parameter Type	Float
Parameter Range	-100dB ~ 100dB
Return	Float
Default	-3 dB
Menu	Marker Fn > N dB BW
Example	:CALCulate:MARK1:BANDwidth:NDB 10

Command Format	:CALCulate:MARKer[1]|2|3|4|5|6|7|8:X:READout FREQuency|TIME|PERiod :CALCulate:MARKer[1]|2|3|4|5|6|7|8:X:READout?
Instruction	Toggles the marker X-Axis readout between frequency, time and period. Gets the marker X-Axis readout type.
Parameter Type	Enumeration
Parameter Range	FREQuency|TIME|PERiod
Return	Enumeration
Default	FREQuency
Menu	Marker Fn > Read Out
Example	:CALCulate:MARKer1:X:READout FREQuency

Command Format	:CALCulate:MARKer:TRCKing[:STATe] OFF|ON|0|1 :CALCulate:MARKer:TRCKing[:STATe]?
Instruction	This command turns on/off signal track state. Gets signal track state.

Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	ON
Menu	Frequency > Signal Track
Example	:CALCulate:MARKer:TRCKing on

4.8 Limit Subsection

:CALCulate:LLINe:TEST:STARt

:CALCulate:LLINe:TEST:STOP

:CALCulate:LLINe:TEST:STATe?

:CALCulate:LLINe[1]|2:STATe

:CALCulate:LLINe[1]|2:TYPE

:CALCulate:LLINe[1]|2:MODE

:CALCulate:LLINe[1]|2:Y

:CALCulate:LLINe[1]|2:DATA

:CALCulate:LLINe[1]|2:ADD

:CALCulate:LLINe[1]|2:DELete

:CALCulate:LLINe[1]|2:ALL:DELete

:CALCulate:LLINe:CONTrol:DOMain

:CALCulate:LLINe:CONTrol:BEEP

:CALCulate:LLINe:FAIL?

:CALCulate:LLINe:FAIL:STOP

:CALCulate:LLINe1|2:OFFSet:X

:CALCulate:LLINe1|2:OFFSet:Y

Command Format	**:CALCulate:LLINe:TEST:STARt**
Instruction	Sets limit test start.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Limit > Test

Example	:CALCulate:LLINe:TEST:STARt

Command Format	**:CALCulate:LLINe:TEST:STOP**
Instruction	Sets limit test stop.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Limit > Test
Example	:CALCulate:LLINe:TEST:STOP

Command Format	**:CALCulate:LLINe:TEST:STATe?**
Instruction	Gets limit test state.
Parameter Type	None
Parameter Range	None
Return	0|1
Default	OFF
Menu	Limit > Test
Example	:CALCulate:LLINe:TEST:STAT?

Command Format	**:CALCulate:LLINe[1]|2:STATe OFF|ON|0|1** **:CALCulate:LLINe[1]|2:STATe?**
Instruction	Sets limit line state. Gets limit line state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	OFF
Menu	Limit > Limit1|2
Example	:CALCulate:LLINe1:STATe OFF

Command Format	**:CALCulate:LLINe[1]|2:TYPE UPPer|LOWer** **:CALCulate:LLINe[1]|2:TYPE?**

Instruction	Mode sets a limit line to be either an upper or lower type limit line. An upper line will be used as the maximum allowable value when comparing with the data. Gets limit type.
Parameter Type	Enumeration
Parameter Range	UPPer|LOWer
Return	Enumeration
Default	The default setting of LINe1 is UPPer, the default setting of LINe2 is LOWer
Menu	Limit > Limit1|2 Edit > Type
Example	:CALCulate:LLINe1: TYPE LOWer

Command Format	**:CALCulate:LLINe[1]|2:MODE LINE|POINt** **:CALCulate:LLINe[1]|2:MODE?**
Instruction	Sets limit mode. Gets limit mode.
Parameter Type	Enumeration
Parameter Range	LINE|POINt
Return	Enumeration
Default	LINE
Menu	Limit > Limit1|2 Edit > Mode
Example	:CALCulate:LLINe1: MODE POINt

Command Format	**:CALCulate:LLINe[1]|2:Y <value>** **:CALCulate:LLINe[1]|2:Y?**
Instruction	Sets the Y-axis value of a limit line. Limit line Y-axis value is set independently and is not affected by the X-axis units. Gets the Y-axis value of a limit line.
Parameter Type	Float
Parameter Range	-400 dBm ~ 330 dBm
Return	Float
Default	0 dBm
Menu	Limit > Limit1|2 Edit > Amplitude
Example	:CALCulate:LLINe1:Y 5dBm

Command Format	**:CALCulate:LLINe[1]|2:DATA <x-axis>,<ampl>{,<x-axis>, <ampl>}** **:CALCulate:LLINe[1]|2:DATA?**
Instruction	Uses this command to define the limit points. Gets the defined limit points.
Parameter Type	X-axis: Float Amplitude: Float

Parameter Range	X-axis: 0 ~ 3.2GHz Amplitude: -400 dBm ~ 330 dBm
Return	X-axis: Float Amplitude: Float
Default	X-axis: -1 Hz Amplitude: 0 dBm
Menu	Limit > Limit1|2 Edit
Example	:CALC:LLINe1:DATA 10000000,-20,20000000,-30

Command Format	**:CALCulate:LLINe[1]|2:ADD <x-axis>,<ampl>**
Instruction	Add limit point data
Parameter Type	X-axis: Float Amplitude: Float
Parameter Range	X-axis: 0 ~ 3.2 G Hz Amplitude: None
Return	X-axis: Float Amplitude: Float
Default	X-axis: -1Hz Amplitude: 0 dBm
Menu	Limit > Limit1|2 Edit
Example	:CALCulate:LLINe1:ADD 10000000,-20

Command Format	**:CALCulate:LLINe[1]|2:DELete <number>**
Instruction	Uses this command to delete the assigned limit point.
Parameter Type	Integer
Parameter Range	None
Return	None
Default	None
Menu	Limit > Limit1|2 Edit > Del Point
Example	:CALCulate:LLINe1:DELete 2

Command Format	**:CALCulate:LLINe[1]|2:ALL:DELete**
Instruction	Uses this command to define all the limits points.
Parameter Type	None
Parameter Range	None
Return	None
Default	None

Menu	Limit > Limit1|2 Edit > Del All
Example	:CALCulate:LLINe2:ALL:DELete

Command Format	**:CALCulate:LLINe1|2:OFFSet:X <value>** **:CALCulate:LLINe1|2:OFFSet:X?**
Instruction	Sets the X-axis offset value of a limit line. Gets the X-axis offset value of a limit line.
Parameter Type	Float
Parameter Range	
Return	Float
Default	0dBm
Menu	Limit > Limit1|2 Edit > X Offset
Example	:CALCulate:LLINe1:OFFSet:X 200

Command Format	**:CALCulate:LLINe1|2:OFFSet:Y <value>** **:CALCulate:LLINe1|2:OFFSet:Y?**
Instruction	Sets the Y-axis offset value of a limit line. Gets the Y-axis offset value of a limit line.
Parameter Type	Float
Parameter Range	-350 dBm ~ 380 dBm
Return	Float
Default	0dBm
Menu	Limit > Limit1|2 Edit > Ampt Offset
Example	:CALCulate:LLINe1:OFFSet:Y 5dBm

Command Format	**:CALCulate:LLINe:CONTrol:DOMain FREQuency|TIME** **:CALCulate:LLINe:CONTrol:DOMain?**
Instruction	Toggles the limit X-axis value between frequency and time. Gets the limit X-axis unit.
Parameter Type	Enumeration
Parameter Range	FREQuency|TIME
Return	Enumeration
Default	FREQuency
Menu	Limit > Setup > X Axis
Example	:CALCulate:LLINe:CONTrol:DOMain FREQuency

Command Format	**:CALCulate:LLINe:CONTrol:BEEP OFF|ON|0|1** **:CALCulate:LLINe:CONTrol:BEEP?**
Instruction	Use this command to turn on/off the limit beep status. Gets limit beep state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	ON
Menu	Limit > Setup > Buzzer
Example	:CALCulate:LLINe:CONTrol:BEEP OFF

Command Format	**:CALCulate:LLINe:FAIL?**
Instruction	This query returns the limits pass/failed result. If the test result fails, this command will get result FAIL. If the test result passes, it will get result PASS.
Parameter Type	None
Parameter Range	None
Return	PASS|FAIL
Default	None
Menu	None
Example	:CALCulate:LLINe:FAIL?

Command Format	**:CALCulate:LLINe:FAIL:STOP OFF|ON|0|1** **:CALCulate:LLINe:FAIL:STOP?**
Instruction	Sets whether to stop the test if the test fails. Gets whether to stop the test if the test fails.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	OFF
Menu	Limit > Setup > Fail to stop
Example	:CALCulate:LLINe:FAIL:STOP OFF

4.9 Measurement Subsystem

Reflection Subsection

Reflection Subsection

SSA series products support this function; SVA series products do not support this function.

[:SENSe]:CAT:RST

[:SENSe]:CAT:FREFlect:TYPE

[:SENSe]:CAT:FREFlect:OPEN

[:SENSe]:CAT:FREFlect:SHORt

[:SENSe]:CAT:FREFlect:LOAD

[:SENSe]:MEASure:REFLction[1]|2|3|4|5|6|7|8:RETUrnloss?

[:SENSe]:MEASure:REFLction[1]|2|3|4|5|6|7|8:COEFficient?

[:SENSe]:MEASure:REFLction[1]|2|3|4|5|6|7|8:VSWR?

Command Format	**[:SENSe]:CAT:RST**
Instruction	Clears calibration data.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Meas > Reflection > Meas Setup > Calibration > Reset
Example	INSTrument:MEASure REFL :CAT:RST

Command Format	**[:SENSe]:CAT:FREFlect:TYPE** **[:SENSe]:CAT:FREFlect:TYPE?**
Instruction	Sets calibration type. Gets calibration type.
Parameter Type	Enumeration
Parameter Range	OPEN:open HOS:(open+short)/2 OL:open+load
Return	Enumeration
Default	None
Menu	Meas > Reflection > Meas Setup > Calibration
Example	INSTrument:MEASure REFL :CAT:FREFlect:TYPE OL

Command Format	**[:SENSe]:CAT:FREFlect:OPEN**
Instruction	Calibration open circuit.
Parameter Type	None
Parameter Range	None

Return	None
Default	None
Menu	Meas > Reflection > Meas Setup > Calibration
Example	INSTrument:MEASure REFL :CAT:FREFlect:OPEN

Command Format	**[:SENSe]:CAT:FREFlect:SHORt**
Instruction	Calibration short circuit.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Meas > Reflection > Meas Setup > Calibration
Example	INSTrument:MEASure REFL :CAT:FREFlect:SHOR

Command Format	**[:SENSe]:CAT:FREFlect:LOAD**
Instruction	Calibration load circuit
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Meas > Reflection > Meas Setup > Calibration
Example	INSTrument:MEASure REFL :CAT:FREFlect:LOAD

Command Format	**[:SENSe]:MEASure:REFLction[1]|2|3|4|5|6|7|8:RETUrnloss?**
Instruction	Reads return loss of reflection measurement.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Meas > Reflection
Example	:MEASure:REFLction2:RETUrnloss?

Command Format	[:SENSe]:MEASure:REFLction[1]|2|3|4|5|6|7|8:COEFficient?
Instruction	Reads Refl coefficient of reflection measurement.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Meas > Reflection
Example	:MEASure:REFLction2:COEF?

Command Format	[:SENSe]:MEASure:REFLction[1]|2|3|4|5|6|7|8:VSWR?
Instruction	Reads VSWR of reflection measurement.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Meas > Reflection
Example	:MEASure:REFLction2:VSWR?

ACPR

CHP

OBW

T-Power

SPECtrogram

TOI

CNR

Harmonics

:INSTrument:MEASure

Command Format	:INSTrument:MEASure OFF|ACPR|CHPower|OBW|TPOWer |SPECtrogram|TOI :INSTrument:MEASure?
Instruction	Sets measure mode. Gets measure mode.
Parameter Type	Enumeration
Parameter Range	OFF: measure off REFLection: Reflection // SSA series products support this function; ACPR: ACPR CHPower: Channel Power

	OBW: Occupied BW TPOWer: T-POWer SPECtrogram: Spectrogram Monitor TOI: Third-order Intercept Point HARMonics: Harmonic analysis CNR: Carrier Noise Ratio
Return	Enumeration
Default	OFF
Menu	Measure
Example	:INSTrument:MEASure ACPR

4.9.1 Reflection Subsection

SSA series products support this function; SVA series products do not support this function.

[:SENSe]:CAT:RST

[:SENSe]:CAT:FREFlect:TYPE

[:SENSe]:CAT:FREFlect:OPEN

[:SENSe]:CAT:FREFlect:SHORt

[:SENSe]:CAT:FREFlect:LOAD

[:SENSe]:MEASure:REFLction[1]|2|3|4|5|6|7|8:RETUrnloss?

[:SENSe]:MEASure:REFLction[1]|2|3|4|5|6|7|8:COEFficient?

[:SENSe]:MEASure:REFLction[1]|2|3|4|5|6|7|8:VSWR?

Command Format	**[:SENSe]:CAT:RST**
Instruction	Clears calibration data.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Meas > Reflection > Meas Setup > Calibration > Reset
Example	INSTrument:MEASure REFL :CAT:RST

Command Format	**[:SENSe]:CAT:FREFlect:TYPE** **[:SENSe]:CAT:FREFlect:TYPE?**
Instruction	Sets calibration type. Gets calibration type.
Parameter Type	Enumeration
Parameter Range	OPEN:open HOS:(open+short)/2

	OL:open+load
Return	Enumeration
Default	None
Menu	Meas > Reflection > Meas Setup > Calibration
Example	INSTrument:MEASure REFL :CAT:FREFlect:TYPE OL

Command Format	**[:SENSe]:CAT:FREFlect:OPEN**
Instruction	Calibration open circuit.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Meas > Reflection > Meas Setup > Calibration
Example	INSTrument:MEASure REFL :CAT:FREFlect:OPEN

Command Format	**[:SENSe]:CAT:FREFlect:SHORt**
Instruction	Calibration short circuit.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Meas > Reflection > Meas Setup > Calibration
Example	INSTrument:MEASure REFL :CAT:FREFlect:SHOR

Command Format	**[:SENSe]:CAT:FREFlect:LOAD**
Instruction	Calibration load circuit
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Meas > Reflection > Meas Setup > Calibration
Example	INSTrument:MEASure REFL :CAT:FREFlect:LOAD

Command Format	[:SENSe]:MEASure:REFLction[1]|2|3|4|5|6|7|8:RETUrnloss?
Instruction	Reads return loss of reflection measurement.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Meas > Reflection
Example	:MEASure:REFLction2:RETUrnloss?

Command Format	[:SENSe]:MEASure:REFLction[1]|2|3|4|5|6|7|8:COEFficient?
Instruction	Reads Refl coefficient of reflection measurement.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Meas > Reflection
Example	:MEASure:REFLction2:COEF?

Command Format	[:SENSe]:MEASure:REFLction[1]|2|3|4|5|6|7|8:VSWR?
Instruction	Reads VSWR of reflection measurement.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Meas > Reflection
Example	:MEASure:REFLction2:VSWR?

4.9.2 ACPR Subsection

[:SENSe]:ACPRatio:BWIDth:INTegration

[:SENSe]:ACPRatio:OFFSet:BWIDth[:INTegration]

[:SENSe]:ACPRatio:OFFSet[:FREQuency]

:MEASure:ACPRatio:MAIN?

:MEASure:ACPRatio:LOWer:POWer?

:MEASure:ACPRatio:LOWer?

:MEASure:ACPRatio:UPPer:POWer?

:MEASure:ACPRatio:UPPer?

Command Format	**[:SENSe]:ACPRatio:BWIDth:INTegration <freq>** **[:SENSe]:ACPRatio:BWIDth:INTegration?**
Instruction	Specifies the range of integration used in calculating the power in the main channel. Gets the range of integration used in calculating the power in the main channel.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	100 Hz ~ 1.5 GHz
Return	Float, unit: Hz
Default	1MHz
Menu	Meas > ACPR > Meas Setup > Main Channel
Example	INSTrument:MEASure ACPR :ACPRatio:BWIDth:INTegration 20 MHz

Command Format	**[:SENSe]:ACPRatio:OFFSet:BWIDth[:INTegration] <freq>** **[:SENSe]:ACPRatio:OFFSet:BWIDth[:INTegration]?**
Instruction	Specifies the bandwidth used in calculating the power in the adjacent channel. Gets the bandwidth used in calculating the power in the adjacent channel.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	100 Hz ~ 1.5 GHz
Return	Float, unit: Hz
Default	1 MHz
Menu	Meas > ACPR > Meas Setup > Adjacent Chn
Example	:ACPRatio:OFFSet:BWIDth 20 MHz

Command Format	**[:SENSe]:ACPRatio:OFFSet[:FREQuency] <freq>** **[:SENSe]:ACPRatio:OFFSet[:FREQuency]?**
Instruction	Sets the space value between the center frequency of main channel power and that of the adjacent channel power. Gets adjacent channel space
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	100 Hz ~ 700 MHz
Return	Float, unit: Hz
Default	3MHz

Menu	Meas > ACPR > Meas Setup > Adj Chn Space
Example	:ACPRatio:OFFSets 20 MHz

Command Format	**:MEASure:ACPRatio:MAIN?**
Instruction	Returns the main channel power of ACPR measurement.
Parameter Type	None
Parameter Range	None
Return	Float, unit: dBm
Default	None
Menu	Meas > ACPR
Example	:MEASure:ACPRatio:MAIN?

Command Format	**:MEASure:ACPRatio:LOWer:POWer?**
Instruction	Returns the lower adjacent channel power of ACPR measurement.
Parameter Type	None
Parameter Range	None
Return	Float, unit: dBm
Default	None
Menu	Meas > ACPR
Example	:MEASure:ACPRatio:LOWer:POWer?

Command Format	**:MEASure:ACPRatio:LOWer?**
Instruction	Returns the lower adjacent channel power to main channel power ratio.
Parameter Type	None
Parameter Range	None
Return	Float, unit: dBm
Default	None
Menu	Meas > ACPR
Example	:MEASure:ACPRatio:LOWer?

Command Format	**:MEASure:ACPRatio:UPPer:POWer?**
Instruction	Returns the upper adjacent channel power of ACPR measurement.
Parameter Type	None
Parameter Range	None
Return	Float, unit: dBm
Default	None
Menu	Meas > ACPR
Example	:MEASure:ACPRatio:UPPer:POWer?

Command Format	**:MEASure:ACPRatio:UPPer?**
Instruction	Returns the upper adjacent channel power to main channel power ratio.
Parameter Type	None
Parameter Range	None
Return	Float
Default	None
Menu	Meas > ACPR
Example	:MEASure:ACPRatio:UPPer?

4.9.3 CHP Subsection

[:SENSe]:CHPower:BWIDth:INTegration

[:SENSe]:CHPower:FREQuency:SPAN:POWer

:MEASure:CHPower?

:MEASure:CHPower:CHPower?

:MEASure:CHPower:DENSity?

Command Format	**[:SENSe]:CHPower:BWIDth:INTegration <freq>** **[:SENSe]:CHPower:BWIDth:INTegration?**
Instruction	Specifies the integration bandwidth to calculate the power. Gets the integration bandwidth.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	100 Hz ~ 3.2 GHz
Return	Float, unit: Hz
Default	2 MHz
Menu	Meas > Ch Power > Meas Setup > Integration BW

Example	:CHPower:BWIDth:INTegration 1.0 GHz

Command Format	**[:SENSe]:CHPower:FREQuency:SPAN:POWer**
Instruction	Sets the analyzer span for the channel power measurement. Be sure the span is set larger than the integration bandwidth.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Meas > Ch Power > Meas Setup > Span Power
Example	:CHPower:FREQuency:SPAN:POWer

Command Format	**:MEASure:CHPower?**
Instruction	This command returns scalar results of main channel power, and power density.
Parameter Type	None
Parameter Range	None
Return	Float, Channel Power unit: dBm Float, Density unit: dBm/Hz
Default	None
Menu	Meas > Ch Power
Example	:MEASure:CHPower?

Command Format	**:MEASure:CHPower:CHPower?**
Instruction	This command returns the value of the channel power in dBm units.
Parameter Type	None
Parameter Range	None
Return	Float
Default	None
Menu	Meas > Ch Power
Example	:MEASure:CHPower:CHPower?

Command	**:MEASure:CHPower:DENSity?**

Format	
Instruction	This command returns the value of the channel power density in dBm/Hz.
Parameter Type	None
Parameter Range	None
Return	Float
Default	None
Menu	Meas > Ch Power
Example	:MEASure:CHPower:DENSity?

4.9.4 OBW Subsection

[:SENSe]:OBWidth:METHod

[:SENSe]:OBWidth:PERCent

[:SENSe]:OBWidth:XDB

:MEASure:OBWidth?

:MEASure:OBWidth:OBWidth?

:MEASure:OBWidth:CENTroid?

:MEASure:OBWidth:OBWidth:FERRor?

Command Format	**[:SENSe]:OBWidth:METHod PERCent|DBC** **[:SENSe]:OBWidth:METHod?**
Instruction	This command toggles the method of OBW measurement between percent and dBc. Gets the method of OBW measurement.
Parameter Type	Enumeration
Parameter Range	PERCent|DBC
Return	Enumeration
Default	PERCent
Menu	Meas > Occupied BW > Meas Setup > Method
Example	:OBW:METHod PERCent

Command Format	**[:SENSe]:OBWidth:PERCent <para>** **[:SENSe]:OBWidth:PERCent?**
Instruction	Edit the percentage of signal power used when determining the occupied bandwidth. Press {%} to set the percentage ranging from 10.00% to 99.99%. Gets the percentage of signal power.
Parameter Type	Float
Parameter Range	10~99.99
Return	Float

Default	99
Menu	Meas > Occupied BW > Meas Setup > %
Example	:OBW:PERCent 50

Command Format	**[:SENSe]:OBWidth:XDB <value>** **[:SENSe]:OBWidth:XDB?**
Instruction	Specify the power level used to determine the emission bandwidth as the number of dB down from the highest signal point, within the occupied bandwidth span. Gets dBc value.
Parameter Type	Float
Parameter Range	0.1 ~ 100
Return	Float
Default	26
Menu	Meas > Occupied BW > Meas Setup > dBc
Example	:OBWidth:XDB 3

Command Format	**:MEASure:OBWidth?**
Instruction	Uses this command to query the occupied bandwidth and bandwidth centroid according to the method you set.
Parameter Type	None
Parameter Range	None
Return	Float, unit: Hz
Default	None
Menu	Meas > Occupied BW
Example	:MEASure:OBW?

Command Format	**:MEASure:OBWidth:OBWidth?**
Instruction	Uses this command to query the occupied bandwidth according to the method you set. Query Centroid Result.
Parameter Type	None
Parameter Range	None
Return	Float, unit: Hz
Default	None
Menu	Meas > Occupied BW
Example	:MEASure:OBW:OBW?

Command Format	**:MEASure:OBWidth:CENTroid?**
Instruction	Uses this command to query the occupied bandwidth according to the method you set.
Parameter Type	None
Parameter Range	None
Return	Float, unit: Hz
Default	None
Menu	Meas > Occupied BW
Example	: MEASure:OBW:CENTroid?

Command Format	**:MEASure:OBWidth:OBWidth:FERRor?**
Instruction	Uses this command to query transmit frequency error.
Parameter Type	None
Parameter Range	None
Return	Float, unit: Hz
Default	None
Menu	Meas > Occupied BW
Example	:MEASure:OBWidth:OBWidth:FERRor?

4.9.5 SubsectionT-power(T-Power)

[:SENSe]:TPOWer:FREQuency:CENTer

[:SENSe]:TPOWer:LLIMit

[:SENSe]:TPOWer:RLIMit

:MEASure:TPOWer?

Command Format	**[:SENSe]:TPOWer:FREQuency:CENTer <freq>** **[:SENSe]:TPOWer:FREQuency:CENTer?**
Instruction	Sets T-power center frequency. Gets T-power center frequency.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	0 ~ 3.2 GHz
Return	Float, unit: Hz
Default	1.5 GHz

Menu	Meas > T-power > Meas Setup > Center Freq
Example	:TPOWer:FREQuency:CENTer 15kHz

Command Format	**[:SENSe]:TPOWer:LLIMit <time>** **[:SENSe]:TPOWer:LLIMit?**
Instruction	Sets T-power start line. Gets T-power start line.
Parameter Type	Float, unit: s
Parameter Range	0 ~ 1000 s
Return	Float, unit: s
Default	0
Menu	Meas > T-power > Meas Setup > Start Line
Example	:TPOWer:LLIMit 0.01

Command Format	**[:SENSe]:TPOWer:RLIMit <time>** **[:SENSe]:TPOWer:RLIMit?**
Instruction	Sets T-power stop line. Gets T-power stop line.
Parameter Type	Float, unit: s
Parameter Range	0 ~ 1000 s
Return	Float, unit: s
Default	20 ms
Menu	Meas > T-power > Meas Setup > Stop Line
Example	:TPOWer:RLIMit 0.02

Command Format	**:MEASure:TPOWer?**
Instruction	Querys the result of T-power measurement.
Parameter Type	Float, unit: dBm
Parameter Range	None
Return	Float, unit: dBm
Default	None
Menu	Meas > T-power
Example	:MEASure:TPOWer?

4.9.6 Spectrum Monitor（SPECtrogram）

[:SENSe]:SPECtrogram:STATe

[:SENSe]:SPECtrogram:RESTart

Command Format	**[:SENSe]:SPECtrogram:STATe RUN|PAUSe** **[:SENSe]:SPECtrogram:STATe?**
Instruction	Sets spectrogram state. Gets spectrogram state.
Parameter Type	Enumeration
Parameter Range	RUN: Start PAUSe: Pause
Return	RUN|PAUSe
Default	RUN
Menu	Meas > Spectrum Monitor > Meas Setup > Spectrogram
Example	:SPECtrogram:STATe PAUSe

Command Format	**[:SENSe]:SPECtrogram:RESTart**
Instruction	Restarts spectrogram.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Meas > Spectrum Monitor > Meas Setup > Restart
Example	:SPECtrogram:RESTart

4.9.7 Third-order Intercept Point（TOI）

:MEASure:TOI?

:MEASure:TOI:IP3?

Command Format	**:MEASure:TOI?**
Instruction	Gets the result of Third-order Intercept Point.
Parameter Type	None
Parameter Range	None
Return	Float
Default	None

Menu	Meas > TOI
Example	:MEASure:TOI?

Command Format	**:MEASure:TOI:IP3?**
Instruction	Gets the min intercept of the Lower TOI(Lower 3rd) and the Upper TOI(Upper 3rd).
Parameter Type	None
Parameter Range	None
Return	Float
Default	None
Menu	Meas > TOI
Example	:MEASure:TOI:IP3?

4.9.8 Carrier Noise Ratio(CNR)

[:SENSe]:CNRatio:BANDwidth:INTegration

[:SENSe]:CNRatio:BANDwidth:NOISe

[:SENSe]:CNRatio:OFFSet

:MEASure:CNRatio?

:MEASure:CNRatio:CARRier?

:MEASure:CNRatio:NOISe?

Command Format	**[:SENSe]:CNRatio:BANDwidth:INTegration <freq>** **[:SENSe]:CNRatio:BANDwidth:INTegration?**
Instruction	Sets Carrier BW. Gets Carrier BW.
Parameter Type	Float, Unit: Hz, kHz, MHz, GHz
Parameter Range	100 Hz ~ 6.3999999 GHz
Return	Float, Unit: Hz
Default	3 MHz
Menu	Meas > CNR > Carrier BW
Example	:CNRatio:BANDwidth:INTegration 1.0 GHz

Command Format	**[:SENSe]:CNRatio:BANDwidth:NOISe <freq>** **[:SENSe]:CNRatio:BANDwidth:NOISe?**
Instruction	Sets Noise BW Gets Noise BW
Parameter Type	Float, Unit: Hz, kHz, MHz, GHz

Parameter Range	100 Hz ~ 6.3999999 GHz
Return	Float, Unit: Hz
Default	3 MHz
Menu	Meas > CNR > Noise BW
Example	:CNRatio:BANDwidth:NOISe 1 MHz

Command Format	**[:SENSe]:CNRatio:OFFSet <freq> [:SENSe]:CNRatio:OFFSet?**
Instruction	Sets Freq Offset Gets Freq Offset
Parameter Type	Float, Unit: Hz, kHz, MHz, GHz
Parameter Range	-3.1999999 GHz ~ 3.1999999 GHz
Return	Float, Unit: Hz
Default	3 MHz
Menu	Meas > CNR > Freq Offset
Example	:CNRatio:OFFSet 1 MHz

Command Format	**:MEASure:CNRatio?**
Instruction	Query CNR
Parameter Type	None
Parameter Range	None
Return	Float
Default	None
Menu	Meas > CNR
Example	:MEASure:TOI?

Command Format	**:MEASure:CNRatio:CARRier?**
Instruction	Query Carrier Power
Parameter Type	None
Parameter Range	None
Return	Float
Default	None

Menu	Meas > CNR
Example	:MEASure:CNRatio:CARRier?

Command Format	**:MEASure:CNRatio:NOISe?**
Instruction	Query Noise Power
Parameter Type	None
Parameter Range	None
Return	Float
Default	None
Menu	Meas > CNR
Example	:MEASure:CNRatio:NOISe?

4.9.9 Harmonics(Harmonics)

[:SENSe]:HARMonics:FREQuency:FUNDamental

[:SENSe]:HARMonics:FREQuency:STEP[:INCRement]

[:SENSe]:HARMonics:FREQuency:FUNDamental:AUTO

[:SENSe]:HARMonics:FREQuency:STEP[:INCRement]:AUTO

[:SENSe]:HARMonics:NUMBer

[:SENSe]:HARMonics:SELect

Command Format	**[:SENSe]:HARMonics:FREQuency:FUNDamental <freq>** **[:SENSe]:HARMonics:FREQuency:FUNDamental?**
Instruction	Sets Fundamental Frequency. Gets Fundamental Frequency.
Parameter Type	Float, Unit: Hz, kHz, MHz, GHz
Parameter Range	10 MHz ~ 1.6 GHz
Return	Float, Unit: Hz
Default	
Menu	Meas > Harmonics > Fundamental
Example	:HARMonics:FREQuency:FUNDamental 1.0 GHz

Command Format	**[:SENSe]:HARMonics:FREQuency:STEP[:INCRement] <freq>** **[:SENSe]:HARMonics:FREQuency:STEP[:INCRement]?**
Instruction	Sets Frequency Step. Gets Frequency Step.

Parameter Type	Float, Unit: Hz, kHz, MHz, GHz
Parameter Range	10 MHz ~ 3.19 GHz
Return	Float, Unit:Hz
Default	
Menu	Meas > Harmoniscs > Freq Step
Example	:HARMonics:FREQuency:STEP 1 MHz

Command Format	**[:SENSe]:HARMonics:FREQuency:FUNDamental:AUTO** **[:SENSe]:HARMonics:FREQuency:FUNDamental:AUTO?**
Instruction	Sets Fundamental Freq State. Gets Fundamental Freq State.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	Boolean
Default	1
Menu	Meas > Harmonics > Fundamental
Example	:HARMonics:FREQuency:FUNDamental:AUTO on

Command Format	**[:SENSe]:HARMonics:FREQuency:STEP[:INCRement]:AUTO** **[:SENSe]:HARMonics:FREQuency:STEP[:INCRement]:AUTO?**
Instruction	Sets Freq step State. Gets Freq step State.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	Boolean
Default	1
Menu	Meas > Harmonics > Freq Step
Example	:HARMonics:FREQuency:STEP:AUTO on

Command Format	**[:SENSe]:HARMonics:NUMBer** **[:SENSe]:HARMonics:NUMBer?**
Instruction	Sets Harmonic Number. Gets Harmonic Number.
Parameter Type	Integer
Parameter Range	2 ~ 10
Return	Integer

Default	10
Menu	Meas > Harmonics > Harmonic Num
Example	:HARMonics:NUMBer 2

Command Format	**[:SENSe]:HARMonics:SELect** **[:SENSe]:HARMonics:SELect?**
Instruction	Sets the Harmonic to be selected. Gets the Harmonic which is selected.
Parameter Type	Integer It will set select all Harmonics when the parameter is 0.
Parameter Range	0 ~ 10
Return	Integer
Default	0
Menu	Meas > Harmonics > Select Harmonic
Example	:HARMonics:SELect 4

4.10 TG Subsystem

:OUTPut[:STATe]

:SOURce:POWer[:LEVel][:IMMediate][:AMPLitude]

:SOURce:CORRection:OFFSet

:CALCulate:NTData[:STATe]

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

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

:DISPlay:WINDow:NTTRace[:STATe]

Command Format	**:OUTPut[:STATe] OFF|ON|0|1** **:OUTPut[:STATe]?**
Instruction	Sets TG on or off. Gets TG state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	0
Menu	TG > TG
Example	:OUTPut ON

Command Format	**:SOURce:POWer[:LEVel][:IMMediate][:AMPLitude] <value>** **:SOURce:POWer[:LEVel][:IMMediate][:AMPLitude]?**
Instruction	Sets TG level. Gets TG level.
Parameter Type	Float, unit: dBm
Parameter Range	0 dBm ~ -20 dBm
Return	Float
Default	0 dBm
Menu	TG > TG Level
Example	:SOURce:POWer -20

Command Format	**:SOURce:CORRection:OFFSet <value>** **:SOURce:CORRection:OFFSet?**
Instruction	Sets TG level offsets. Gets TG level offsets.
Parameter Type	Float, unit: dBm
Parameter Range	200 dBm ~ -200 dBm
Return	Float
Default	0 dBm
Menu	TG > TG Level OffSet
Example	:SOURce:CORRection:OFFSet 1

Command Format	**:CALCulate:NTData[:STATe] OFF|ON|0|1** **:CALCulate:NTData[:STATe]?**
Instruction	Sets TG normalize on-off. Gets TG normalize state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	0
Menu	TG > Normalize
Example	:CALCulate:NTData ON

Command Format	**:DISPlay:WINDow:TRACe:Y[:SCALe]:NRLevel <value>** **:DISPlay:WINDow:TRACe:Y[:SCALe]:NRLevel?**
Instruction	Sets TG normalize reference level. Gets TG normalize reference level.
Parameter Type	Float, unit: dB

Parameter Range	-200 dB ~ 200 dB
Return	Float, unit: dB
Default	0 dB
Menu	TG > Normal Ref Level
Example	:DISPlay:WINDow:TRACe:Y:NRLevel 10

Command Format	**:DISPlay:WINDow:TRACe:Y[:SCALe]:NRPosition <integer>** **:DISPlay:WINDow:TRACe:Y[:SCALe]:NRPosition?**
Instruction	Sets TG normalize reference position. Gets TG normalize reference position.
Parameter Type	Integer
Parameter Range	0 ~ 100%
Return	Float
Default	100%
Menu	TG > Normal Ref Pos
Example	:DISPlay:WINDow:TRACe:Y:NRPosition 10

Command Format	**:DISPlay:WINDow:NTTRace[:STATe] OFF|ON|0|1** **:DISPlay:WINDow:NTTRace[:STATe]?**
Instruction	Sets TG normalize reference trace on or off. Gets the state of TG normalize reference trace.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	0
Menu	TG > Ref Trace
Example	:DISPlay:WINDow:NTTRace ON

4.11 Demod Subsystem

Command Format	**[:SENSe]:DEMod AM|FM|OFF** **[:SENSe]:DEMod?**

Instruction	Sets demod mode. Gets demod mode.
Parameter Type	Enumeration
Parameter Range	AM|FM|OFF
Return	Enumeration
Default	OFF
Menu	Demod
Example	:DEMod AM

Command Format	[:SENSe]:DEMod:TIME <time> [:SENSe]:DEMod:TIME?
Instruction	Sets demod time. Gets demod time.
Parameter Type	Float, unit: ms, us, s
Parameter Range	5 ms ~1000 s
Return	Float, unit: s
Default	5 ms
Menu	Demod
Example	DEMod:TIME 5 ms

Command Format	[:SENSe]:DEMod:EPHone OFF|ON|0|1 [:SENSe]:DEMod:EPHone?
Instruction	Sets earphone on-off. Gets earphone on-off.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	OFF
Menu	Demod > Earphone
Example	:DEMod:EPHone ON

Command Format	[:SENSe]:DEMod:VOLume <value> [:SENSe]:DEMod:VOLume?
Instruction	Sets volume value. Gets volume value.
Parameter Type	Integer
Parameter Range	0 ~ 10

Return	Integer
Default	6
Menu	Demod > Volume
Example	:DEMod:EPHone ON

5.Vector Network Analyzer

The ‘ch’ in Command is a channel parameter in VNA mode, only supports 1, and can be omitted. That is used to be compatible with the command format of other companies. In most cases, it can be ignored.
The Commands in this mode is compatible with Agilent Technologies E5071C series network analyzer

5.1 Frequency Subsection

[:SENSe[ch]]:FREQuency:CENTer

[:SENSe[ch]]:FREQuency:STARt

[:SENSe[ch]]:FREQuency:STOP

[:SENSe[ch]]:FREQuency:SPAN

Command Format	**[:SENSe[ch]]:FREQuency:CENTer <freq>** **[:SENSe[ch]]:FREQuency:CENTer?**
Instruction	Sets the center frequency of VNA. Gets the center frequency.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	100.05 kHz ~ 3.199999950 GHz
Return	Float, unit: Hz
Default	1.60005 GHz
Menu	Frequency > Center Freq
Example	:FREQuency:CENTer 0.2 GHz :SENSe1:FREQuency:CENTer 0.2 GHz :SENSe:FREQuency:CENTer 0.2 GHz

Command Format	**[:SENSe[ch]]:FREQuency:STARt <freq>** **[:SENSe[ch]]:FREQuency:STARt?**
Instruction	Sets the start frequency of VNA. Gets the start frequency.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	100 kHz ~ 3.1999999 GHz
Return	Float, unit: Hz
Default	100 kHz
Menu	Frequency > Start Freq
Example	:FREQuency:STARt 10 MHz

Command Format	**[:SENSe[ch]]:FREQuency:STOP <freq>** **[:SENSe[ch]]:FREQuency:STOP?**
Instruction	Sets the stop frequency of VNA. Gets the stop frequency.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	100.01 kHz ~ 3.2 GHz
Return	Float, unit: Hz
Default	3.2 GHz
Menu	Frequency > Stop Freq
Example	:FREQuency:STOP 1.0 GHz

Command Format	**[:SENSe[ch]]:FREQuency:SPAN <freq>** **[:SENSe[ch]]:FREQuency:SPAN?**
Instruction	Sets the span of VNA. Gets the span frequency.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	100 Hz ~ 3.1999 GHz
Return	Float, unit: Hz
Default	3.1999 GHz
Menu	Span > Span
Example	:FREQuency:SPAN 1 GHz

5.2 Display Subsection

:DISPlay:WINDow[ch]:TRACe[1]|2|3|4:Y[:SCALe]:AUTO

:DISPlay:WINDow[ch]:TRACe[1]|2|3|4:Y[:SCALe]:RLEVel

:DISPlay:WINDow[ch]:TRACe[1]|2|3|4:Y[:SCALe]:PDIVision

:DISPlay:WINDow[ch]:TRACe[1]|2|3|4:Y[:SCALe]:RPOSition

Command Format	**:DISPlay:WINDow[ch]:TRACe[1]|2|3|4:Y[:SCALe]:AUTO**
Instruction	Sets auto scale.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Amplitude > Auto Scale
Example	:DISPlay:WINDow1:TRACe2:Y:SCALe:AUTO

Command Format	**:DISPlay:WINDow[ch]:TRACe[1]|2|3|4:Y[:SCALe]:RLEVel <value>** **:DISPlay:WINDow[ch]:TRACe[1]|2|3|4:Y[:SCALe]:RLEVel?**
Instruction	Sets reference level. Gets reference level.
Parameter Type	Float Unit: dB, time units(s, ms, us, ps)
Parameter Range	-1000 dB ~ 1000 dB
Return	Float
Default	0 dBm
Menu	Amplitude > Ref Level
Example	:DISPlay:WINDow:TRACe:Y:RLEVel 20 DB

Command Format	**:DISPlay:WINDow[ch]:TRACe[1]|2|3|4:Y[:SCALe]:PDIVision <integer>** **:DISPlay:WINDow[ch]:TRACe[1]|2|3|4:Y[:SCALe]:PDIVision?**
Instruction	This command sets the per-division display scaling for the y-axis. Gets Scale/Div
Parameter Type	Float, Unit: dB, time units (s, ms, us, ps)
Parameter Range	0.1 dB ~ 1000 dB
Return	Float
Default	10 dB
Menu	Amplitude > Scale/Div
Example	:DISPlay:WINDow:TRACe:Y:PDIVision 10 dB

Command Format	**:DISPlay:WINDow[ch]:TRACe[1]|2|3|4:Y[:SCALe]:RPOSition <integer>** **:DISPlay:WINDow[ch]:TRACe[1]|2|3|4:Y[:SCALe]:RPOSition?**
Instruction	Sets Reference Scale Position. Gets Reference Scale Position.
Parameter Type	Integer
Parameter Range	0 ~ 10
Return	Integer
Default	5
Menu	Amplitude > Ref Position
Example	:DISPlay:WINDow:TRACe:Y:SCALe:RPOSition 10

5.3 Bandwidth Subsection

[:SENSe[ch]]:BWIDth[:RESolution]?

Command Format	**[:SENSe[ch]]:BWIDth[:RESolution]?**
Instruction	Querys Intermediate Frequency Bandwidth.
Parameter Type	None
Parameter Range	None
Return	Float, Unit: Hz
Default	10 kHz
Menu	BW > IFBW
Example	:BWIDth?

5.4 Sweep Subsection

[:SENSe[ch]]:SWEep:POINts

:INITiate[ch][:IMMediate]

:INITiate[ch]:CONTinuous

ABORt

Command Format	**[:SENSe[ch]]:SWEep:POINts <integer>** **[:SENSe[ch]]:SWEep:POINts?**
Instruction	Sets sweep points. Gets sweep points.
Parameter Type	Integer

Parameter Range	101 ~ 751
Return	Integer
Default	201
Menu	Sweep > Points
Example	:SWEep:POINts?

Command Format	**:INITiate[ch][:IMMediate]**
Instruction	Restarts the current sweep. :INITiate:RESTart and :INITiate:IMMediate perform exactly the same function.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	
Example	:INITiate:IMMediate

Command Format	**:INITiate[ch]:CONTinuous OFF|ON|0|1** **:INITiate[ch]:CONTinuous?**
Instruction	Sets continuous sweep mode on-off. Gets continuous sweep mode state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	ON
Menu	Sweep > Sweep
Example	:INITiate:CONTinuous OFF

Command Format	**ABORt**
Instruction	This command is used to stop the current measurement. It aborts the current measurement as quickly as possible, resets the sweep and trigger systems, and puts the measurement into an "idle" state. If the analyzer is set for Continuous measurement, it sets up the measurement and initiates a new data measurement sequence with a new data acquisition (sweep) taken once the trigger condition is met. If the analyzer is set for Single measurement, it remains in the "idle" state until an :INIT:IMM command is received.

Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	None
Example	ABORt

5.5 Trace Subsection

:CALCulate[ch]:PARameter[1]2|3|4:DEFine

:CALCulate[ch][:SELected]:FORMat

:CALCulate[ch]:PARameter[1]|2|3|4:SELect

:CALCulate[ch]:PARameter:COUNt

DISP:WINDow[ch]:TRACe[1]|2|3|4:STATe

DISP:WINDow[ch]:TRACe[1]|2|3|4:MEMory[:STATe]

:TRACe[1]|2|3|4:HOLD

:CALCulate[ch][:SELected]:MATH:MEMorize

:CALCulate[ch][:SELected]:MATH:FUNCtion

:CALCulate[ch][:SELected]:DATA:FDATa

:CALCulate[ch][:SELected]:DATA:FMEMory

:FORMat[:TRACe][:DATA]

[:SENSe]:AVERage:TRACe[1]|2|3|4:COUNt

[:SENSe]:AVERage:TRACe1|2|3|4:STATe

[:SENSe[ch]]:AVERage:COUNt

[:SENSe[ch]]:AVERage:STATe

Command Format	**:CALCulate[ch]:PARameter[1]2|3|4:DEFine :CALCulate[ch]:PARameter[1]2|3|4:DEFine?**
Instruction	Sets Measurement parameter. Gets Measurement parameter.
Parameter Type	Enumeration
Parameter Range	S11 S21
Return	Enumeration
Default	S11
Menu	Meas

Example	:CALCulate1:PARameter2:DEFine S11

Command Format	**:CALCulate[ch][:SELected]:FORMat** **:CALCulate[ch][:SELected]:FORMat?**
Instruction	Sets displayed data format of VNA. Querys displayed data format of VNA.
Parameter Type	Enumeration
Parameter Range	MLOGarithmic: Log magnitude PHASe: Phase in degrees GDELay: Group delay SLINear: Smith chart (Lin/Phase) SLOGarithmic: Smith chart (Log/Phase) SCOMplex: Smith chart (Real/Imag) SMITh: Smith chart (R+jX) SADMittance: Smith chart (G+jB) PLINear: Polar chart(Lin/Phase) PLOGarithmic: Polar chart (Log/Phase) POLar: Polar chart (Real/Imag) MLINear: Linear magnitude SWR: Standing Wave Ratio
Return	Enumeration
Default	MLOGarithmic
Menu	Meas > Format
Example	:CALCulate1:FORMat SWR

Command Format	**:CALCulate[ch]:PARameter[1]|2|3|4:SELect**
Instruction	Sets the trace to the current
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Trace > Select Trace
Example	:CALCulate:PARameter2:SELect

Command Format	**:CALCulate[ch]:PARameter:COUNt <integer>** **:CALCulate[ch]:PARameter:COUNt?**
Instruction	Sets trace number. Gets trace number.
Parameter Type	Integer
Parameter Range	1 ~ 4
Return	None

Default	1
Menu	Trace > Num of Trace
Example	:CALCulate:PARameter:COUNt 4

Command Format	**DISP:WINDow[ch]:TRACe[1]|2|3|4:STATe** **DISP:WINDow[ch]:TRACe[1]|2|3|4:STATe?**
Instruction	Turns on/off trace data display state. Gets trace data display state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	ON
Menu	Trace > Display
Example	DISP:WINDow:TRACe2:STATe OFF

Command Format	**DISP:WINDow[ch]:TRACe[1]|2|3|4:MEMory[:STATe]** **DISP:WINDow[ch]:TRACe[1]|2|3|4:MEMory[:STATe]?**
Instruction	Turn on/off trace memory display state. Gets trace memory display state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	0
Menu	Trace > Display
Example	DISP:WINDow[ch]:TRACe[ch]:MEMory OFF

Command Format	**:TRACe[1]|2|3|4:HOLD** **:TRACe[1]|2|3|4:HOLD?**
Instruction	Sets trace hold type. Gets trace hold type.
Parameter Type	Enumeration
Parameter Range	OFF: close trace hold MAX: max hold MIN: min hold
Return	OFF MAX MIN
Default	0
Menu	Trace > Trace Hold

Example	:TRACe2:HOLD MAX

Command Format	**:CALCulate[ch][:SELected]:MATH:MEMorize**
Instruction	Copies trace data to memory.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Trace> Data->Mem
Example	:CALCulate1:SELected:MATH:MEMorize

Command Format	**:CALCulate[ch][:SELected]:MATH:FUNCtion** **:CALCulate[ch][:SELected]:MATH:FUNCtion?**
Instruction	Sets trace math type
Parameter Type	Enumeration
Parameter Range	DIVide MULtiply SUBtract ADD OFF
Return	Enumeration
Default	OFF
Menu	Trace > Math
Example	:CALCulate1:SELected:MATH:MEMorize

Command Format	**:CALCulate[ch][:SELected]:DATA:FDATa** **:CALCulate[ch][:SELected]:DATA:FDATa?**
Instruction	Sets format trace data. Querys format trace data.
Parameter Type	Array data representing NOP (number of measurement points)*2 (formatted data array). N is an integer between 1 and NOP. • Data (n*2-2): Data from the nth measuring point (real part). • Data (n*2-1): data from the nth measurement point (imaginary part). • Array index starts at 0
Parameter Range	
Return	Array data
Default	None
Menu	None

Example	:CALCulate:DATA:FDATa 1,0,0.5,1

Command Format	**:CALCulate[ch][:SELected]:DATA:FMEMory** **:CALCulate[ch][:SELected]:DATA:FMEMory?**
Instruction	Sets format Memory data Query format Memory data
Parameter Type	Array data representing NOP (number of measurement points)*2 (formatted data array). N is an integer between 1 and NOP. • Data (n*2-2): Data from the nth measuring point (real part). • Data (n*2-1): data from the nth measurement point (imaginary part). • Array index starts at 0
Parameter Range	
Return	Memory data
Default	None
Menu	None
Example	:CALCulate:DATA: FMEMory 1,0,0.5,1

Command Format	**:FORMat[:TRACe][:DATA] ASCii|REAL|REAL32** **:FORMat[:TRACe][:DATA]?**
Instruction	Sets trace data type. Gets trace data type.
Parameter Type	Enumeration
Parameter Range	ASCii REAL double precision floating-point (double) REAL32: single precision floating-point (float)
Return	String
Default	ASCii
Menu	None
Example	:FORMat ASCii

Command Format	**[:SENSe]:AVERage:TRACe[1]|2|3|4:COUNt <integer>** **[:SENSe]:AVERage:TRACe[1]|2|3|4:COUNt?**
Instruction	Sets trace average count. Gets trace average count.
Parameter Type	Integer
Parameter Range	1 ~ 999
Return	Integer
Default	100
Menu	Trace > Average
Example	:AVERage:TRACe:COUNt 200

Command Format	**[:SENSe]:AVERage:TRACe1|2|3|4:STATe OFF|ON|0|1** **[:SENSe]:AVERage:TRACe1|2|3|4:STATe?**
Instruction	Sets trace average State. Gets trace average State. It will set the average state of all traces without trace parameters.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	OFF
Menu	Trace > Average
Example	:AVERage:TRACe1:STATe ON

Command Format	**[:SENSe[ch]]:AVERage:COUNt <integer>** **[:SENSe[ch]]:AVERage:COUNt?**
Instruction	Sets trace average count. Gets trace average count(Default Trace 1).
Parameter Type	Integer
Parameter Range	1 ~ 999
Return	Integer
Default	100
Menu	Trace > Average
Example	:AVERage:COUNt 200

Command Format	**[:SENSe[ch]]:AVERage:STATe OFF|ON|0|1** **[:SENSe[ch]]:AVERage:STATe?**
Instruction	Sets all trace average State. Gets all trace average State.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	OFF
Menu	Trace > Average
Example	:AVERage:TRACe1:STATe ON

5.6 Marker Subsection

:CALCulate[ch]:MARKer[1]|2|3|4|5|6|7STATe

:CALCulate[ch]:MARKer[1]|2|3|4|5|6|7:MODE

:CALCulate[ch]:MARKer[1]|2|3|4|5|6|7:X

:CALCulate[ch]:MARKer[1]|2|3|4|5|6|7:Y?

:CALCulate[ch][:SELected]:MARKer:REFerence[:STATe]

:CALCulate[ch][:SELected]:MARKer:DISCrete

:CALCulate:MARKer:AOFF

:CALCulate[ch][:SELected]:MARKer:COUPle

:CALCulate:MARKer[1]|2|3|4|5|6|7:MAXimum

:CALCulate:MARKer[1]|2|3|4|5|6|7:MINimize

:CALCulate:MARKer[1]|2|3|4|5|6|7:CPSearch[:STATe]

:CALCulate:MARKer[1]|2|3|4|5|6|7:CVSearch[:STATe]

:CALCulate:MARKer[1]|2|3|4|5|6|7[:SET]:START

:CALCulate:MARKer[1]|2|3|4|5|6|7[:SET]:STOP

:CALCulate:MARKer[1]|2|3|4|5|6|7[:SET]:CENTer

:CALCulate:MARKer[1]|2|3|4|5|6|7:DELTa[:SET]:SPAN

Command Format	**:CALCulate[ch]:MARKer[1]|2|3|4|5|6|7STATe OFF|ON|0|1** **:CALCulate[ch]:MARKer[1]|2|3|4|5|6|7:STATe?**
Instruction	Sets marker state. Gets marker state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	OFF
Menu	Marker
Example	:CALCulate:MARK1:STATe ON

Command Format	**:CALCulate[ch]:MARKer[1]|2|3|4|5|6|7:MODE POSition|DELTa|OFF** **:CALCulate[ch]:MARKer[1]|2|3|4|5|6|7:MODE?**
Instruction	Sets marker mode. Gets marker mode.
Parameter Type	Enumeration
Parameter Range	POSition DELTa OFF

Return	POS|DELT|OFF
Default	OFF
Menu	Marker
Example	:CALCulate:MARK1:MODE POSition

Command Format	**:CALCulate[ch]:MARKer[1]|2|3|4|5|6|7:X <para>** **:CALCulate[ch]:MARKer[1]|2|3|4|5|6|7:X?**
Instruction	Sets marker X value. Gets marker X value. This command only works when marker is not off.
Parameter Type	Float, Unit: Hz, kHz, MHz, GHz
Parameter Range	100 kHz ~ 3.2 GHz
Return	
Default	1.60005 GHz
Menu	Marker > Normal
Example	:CALCulate:MARKer4:X 0.4 GHz

Command Format	**:CALCulate[ch]:MARKer[1]|2|3|4|5|6|7:Y?**
Instruction	Gets marker Y value
Parameter Type	None
Parameter Range	None
Return	Float
Default	None
Menu	
Example	:CALCulate:MARKer1:Y?

Command Format	**:CALCulate[ch][:SELected]:MARKer:REFerence[:STATe]** **:CALCulate[ch][:SELected]:MARKer:REFerence[:STATe]?**
Instruction	Sets reference marker R state. When set to ON, the other open markers types are changed to Delta, and when OFF, the open markers are set to Normal. Gets reference marker R state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1

Default	OFF
Menu	
Example	:CALCulate:MARKer:REFerence ON

Command Format	**:CALCulate[ch][:SELected]:MARKer:DISCrete OFF|ON|0|1** **:CALCulate[ch][:SELected]:MARKer:DISCrete?**
Instruction	Sets Marker Discrete State (Patterns in which markers move only at measurement points). Gets Marker Discrete State.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	ON
Menu	Marker > Discrete
Example	:CALCulate:SELected:MARKer:DISCrete?

Command Format	**:CALCulate:MARKer:AOFF**
Instruction	Close all markers of current trace.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	None
Example	:CALCulate:MARKer:AOFF

Command Format	**:CALCulate[ch][:SELected]:MARKer:COUPle OFF|ON|0|1** **:CALCulate[ch][:SELected]:MARKer:COUPle?**
Instruction	Sets marker couple state. Gets marker couple state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	OFF
Menu	Marker > Couple
Example	:CALCulate:SELected:MARKer:COUPle?

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7:MAXimum**
Instruction	Performs a peak search in current trace, you can select current trace by using :CALCulate[ch]:PARameter[1]|2|3|4:SELect
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Peak > Peak
Example	:CALCulate:MARKer4:MAXimum

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7:MINimize**
Instruction	Performs a valley search in current trace, you can select current trace by using :CALCulate[ch]:PARameter[1]|2|3|4:SELect
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Peak > Valley
Example	:CALCulate:MARKer4: MINimize

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7:CPSearch[:STATe] OFF|ON|0|1** **:CALCulate:MARKer[1]|2|3|4|5|6|7:CPSearch[:STATe]?**
Instruction	Toggles the continuous peak search function between on and off. Gets the continuous peak search function state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	None
Menu	Peak > Cont Peak
Example	:CALCulate:MARKer1:CPSEarch ON

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7:CVSearch[:STATe] OFF|ON|0|1** **:CALCulate:MARKer[1]|2|3|4|5|6|7:CVSearch[:STATe]?**
Instruction	Toggles the continuous valley search function between on and off.

	Gets the continuous valley search function state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	None
Menu	Peak > Cont Valley
Example	:CALCulate:MARKer1:CVSEarch ON

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7[:SET]:START**
Instruction	Sets the start frequency to the value of the specified marker frequency. This command is valid when the Marker is on.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Marker→ > M→Start Freq
Example	:CALCulate:MARKer1:START

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7[:SET]:STOP**
Instruction	Sets the stop frequency to the value of the specified marker frequency. This command is valid when the Marker is on.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Marker→ > Marker→Stop Freq
Example	:CALCulate:MARKer1:STOP

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7[:SET]:CENTer**
Instruction	This command sets the center frequency equal to the specified marker frequency. This command is valid when the Marker is on.
Parameter Type	None
Parameter Range	None

Return	None
Default	None
Menu	Marker→ > M→CF
Example	:CALCulate:MARKer1:CENTer

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7:DELTa[:SET]:SPAN**
Instruction	This command sets the span equal to the specified delta marker frequency. This command can be only used in DELTa marker mode, Reference Command:CALCulate[ch]:MARKer[1]|2|3|4|5|6|7:MODE
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Marker→ > △ M→Span
Example	:CALCulate:MARKer2:DELTa:SPAN

5.7 Calibration Subsystem

[:SENSe[ch]]:CORRection:RVELocity:COAX

[:SENSe[ch]]:CORRection:COLLect[:ACQuire]:LOAD

[:SENSe[ch]]:CORRection:COLLect[:ACQuire]:OPEN

[:SENSe[ch]]:CORRection:COLLect[:ACQuire]:SHORt

[:SENSe[ch]]:CORRection:COLLect[:ACQuire]:THRU

[:SENSe[ch]]:CORRection:COLLect:CKIT:LABel

[:SENSe]:CORRection:COLLect:CKIT:GENDer

[:SENSe[ch]]:CORRection:COLLect:CKIT:LABel:CATalog?

[:SENSe[ch]]:CORRection:COLLect:METHod:SOLT1

[:SENSe[ch]]:CORRection:COLLect:METHod[:RESPonse]:THRU

[:SENSe[ch]]:CORRection:COLLect:METHod:TYPE?

[:SENSe[ch]]:CORRection:COLLect:CLEar

[:SENSe[ch]]:CORRection:COLLect:SAVE

Command Format	**[:SENSe[ch]]:CORRection:RVELocity:COAX** **[:SENSe[ch]]:CORRection:RVELocity:COAX?**

Instruction	Sets Velocity Factor. Gets Velocity Factor.
Parameter Type	Float
Parameter Range	0.1 ~ 1
Return	Float
Default	0.66
Menu	Meas > Calibration > Velocity Factor
Example	:CORRection:RVELocity:COAX 0.5

Command Format	**[:SENSe[ch]]:CORRection:COLLect[:ACQuire]:LOAD**
Instruction	Measures the Load calibration standard that is connected to the specified port.
Parameter Type	Integer
Parameter Range	1 (meas port1)
Return	None
Default	None
Menu	Meas > Cailbration > Calibrate > 1-Port Cal
Example	:CORRection:COLLect:LOAD 1

Command Format	**[:SENSe[ch]]:CORRection:COLLect[:ACQuire]:OPEN**
Instruction	Measures the OPEN calibration standard that is connected to the specified port.
Parameter Type	Integer
Parameter Range	1(meas port1)
Return	None
Default	None
Menu	Meas > Cailbration > Calibrate > 1-Port Cal
Example	:CORRection:COLLect:OPEN 1

Command Format	**[:SENSe[ch]]:CORRection:COLLect[:ACQuire]:SHORt**
Instruction	Measures the Short calibration standard that is connected to the specified port.
Parameter Type	Integer
Parameter Range	1

Return	None
Default	None
Menu	Meas > Cailbration > Calibrate > 1-Port Cal
Example	:CORRection:COLLect:SHOR 1

Command Format	**[:SENSe[ch]]:CORRection:COLLect[:ACQuire]:THRU**
Instruction	Measures the THRU calibration standard that is connected between the specified ports.
Parameter Type	Integer (Port1 and Port2)
Parameter Range	1, 2
Return	None
Default	None
Menu	Meas > Cailbration > Calibrate > ResponseThrough
Example	:CORRection:COLLect:THRU 1,2

Command Format	**[:SENSe[ch]]:CORRection:COLLect:CKIT:LABel** **[:SENSe[ch]]:CORRection:COLLect:CKIT:LABel?**
Instruction	Sets the Cal Kit to use. Gets the Cal Kit.
Parameter Type	String (you should use “” when you input kits name)
Parameter Range	"F503ME", ”85032F”,” 85036B/E”,”User1”,”User2”
Return	"F503ME", ”85032F”,” 85036B/E”, ”User1”,”User2”
Default	"F503ME"
Menu	Meas > Cailbration > Cal Kit
Example	:CORRection:COLLect:CKIT:LABel ”85032F”

Command Format	**[:SENSe]:CORRection:COLLect:CKIT:GENDer** **[:SENSe]:CORRection:COLLect:CKIT:GENDer？**
Instruction	Setsthe gender of Calibration Kits. Gets the gender of Calibration Kits.
Parameter Type	String
Parameter Range	Male Female
Return	Male Female
Default	Male
Menu	Meas > Cailbration > Cal Kit

Example	:CORRection:COLLect:CKIT:GENDer Female

Command Format	**[:SENSe[ch]]:CORRection:COLLect:CKIT:LABel:CATalog?**
Instruction	Reads the available Cal Kits in the SVA1000.
Parameter Type	None
Parameter Range	None
Return	"F503ME", ”85032F”,” 85036B/E”
Default	
Menu	
Example	:CORRection:COLLect:CKIT:LABel:CATalog?

Command Format	**[:SENSe[ch]]:CORRection:COLLect:METHod:SOLT1**
Instruction	Sets the Cal Method to 1-port SOLT calibration.
Parameter Type	Integer
Parameter Range	1
Return	None
Default	None
Menu	Meas > Cailbration > Calibrate > 1-Port Cal
Example	:CORRection:COLLect:METHod:SOLT1 1

Command Format	**[:SENSe[ch]]:CORRection:COLLect:METHod[:RESPonse]:THRU**
Instruction	Sets the Cal Method to 2-port TRL calibration.
Parameter Type	Integer
Parameter Range	Port(1, 2)
Return	None
Default	None
Menu	Meas > Cailbration > Calibrate > ResponseThrough
Example	:CORRection:COLLect:METHod:THRU 1,2

Command Format	[:SENSe[ch]]:CORRection:COLLect:METHod:TYPE?
Instruction	Querys Calibration type.
Parameter Type	None
Parameter Range	None
Return	"NONE", " SOLT1", " RESPT" (Enheneed Response Not added yet)
Default	
Menu	
Example	:CORRection:COLLect:METHod:TYPE?

Command Format	[:SENSe[ch]]:CORRection:COLLect:CLEar
Instruction	Clears Calibration Data.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	None
Example	:CORRection:COLLect:CLEar

Command Format	[:SENSe[ch]]:CORRection:COLLect:SAVE
Instruction	Ends the calibration
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	None
Example	:CORRection:COLLect:SAVE

5.8 Port Extensions

[:SENSe[ch]]:CORRection:EXTension[:STATe]

[:SENSe[ch]]:CORRection:EXTension:PORT[1]2:TIME

[SENSe[Ch]]:CORRection:EXTension:AUTO:PORT

Command Format	**[:SENSe[ch]]:CORRection:EXTension[:STATe]** **[:SENSe[ch]]:CORRection:EXTension[:STATe]?**
Instruction	Sets port extension state. Gets port extension state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	OFF
Menu	Meas > Cailbration > Port Extensions
Example	:CORRection:EXTension ON

Command Format	**[:SENSe[ch]]:CORRection:EXTension:PORT[1]2:TIME** **[:SENSe[ch]]:CORRection:EXTension:PORT[1]2:TIME?**
Instruction	Sets extended port delay. Gets extended port delay.
Parameter Type	Float
Parameter Range	
Return	Float
Default	0
Menu	Meas > Cailbration > Port Extensions > Delay Port1|2
Example	:CORRection:EXTension:PORT1:TIME

Command Format	**[SENSe[Ch]]:CORRection:EXTension:AUTO:PORT**
Instruction	At present, only automatic open port 1 is supported, and off is not processed
Parameter Type	Boolean
Parameter Range	Boolean
Return	None
Default	None

Menu	Meas > Cailbration > Port Extensions > Auto Open Port1
Example	:CORRection:EXTension:AUTO:PORT 1,ON

6.Distance To Fault

6.1 Frequency Subsection

[:SENSe]:FREQuency:CENTer

[:SENSe]:FREQuency:STARt

[:SENSe]:FREQuency:STOP

[:SENSe]:FREQuency:SPAN

Command Format	**[:SENSe]:FREQuency:CENTer <freq>** **[:SENSe]:FREQuency:CENTer?**
Instruction	Sets the center frequency. Gets the center frequency.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	100.05 kHz ~ 3.199999950 GHz
Return	Float, unit: Hz
Default	1.60005 GHz
Menu	Frequency > Center Freq
Example	:FREQuency:CENTer 0.2 GHz :SENSe:FREQuency:CENTer 0.2 GHz

Command Format	**[:SENSe]:FREQuency:STARt <freq>** **[:SENSe]:FREQuency:STARt?**
Instruction	Sets the start frequency. Gets the start frequency.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	100 kHz ~ 3.1999999 GHz
Return	Float, unit: Hz

Default	100 kHz
Menu	Frequency > Start Freq
Example	:FREQuency:STARt 10 MHz

Command Format	**[:SENSe]:FREQuency:STOP <freq>** **[:SENSe]:FREQuency:STOP?**
Instruction	Sets the stop frequency. Gets the stop frequency.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	100.01 kHz ~ 3.2 GHz
Return	Float, unit: Hz
Default	3.2 GHz
Menu	Frequency > Stop Freq
Example	:FREQuency:STOP 1.0 GHz

Command Format	**[:SENSe]:FREQuency:SPAN <freq>** **[:SENSe]:FREQuency:SPAN?**
Instruction	Sets the span frequency. Gets the span frequency.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	100 Hz ~ 3.1999 GHz
Return	Float, unit: Hz
Default	3.1999 GHz
Menu	Span > Span
Example	:FREQuency:SPAN 1 GHz

6.2 Amplitude Subsection

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

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

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

Command Format	**:DISPlay:WINDow:TRACe:Y[:SCALe]:AUTO**
Instruction	Sets auto scale.
Parameter Type	None
Parameter	None

Range	
Return	None
Default	None
Menu	Amplitude > Auto Scale
Example	:DISPlay:WINDow:TRACe:Y:SCALe:AUTO

Command Format	**:DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel <value>** **:DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel?**
Instruction	Sets reference level. Gets reference level.
Parameter Type	Float
Parameter Range	When meas type is CAT: 0.5~100 When meas type is DTF: -10~10
Return	Float
Default	1
Menu	Amplitude > Ref Level
Example	:DISPlay:WINDow:TRACe:Y:RLEVel 2

Command Format	**:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision <integer>** **:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision?**
Instruction	This command sets the per-division display scaling for the y-axis. Gets Scale/Div.
Parameter Type	Float
Parameter Range	When meas type is CAT: 0.1~10 When meas type is DTF: 0.01~10
Return	Float
Default	1 When meas type is CAT:1 When meas type is DTF:5
Menu	Amplitude > Scale/Div
Example	:DISPlay:WINDow:TRACe:Y:PDIVision 2

6.3 Sweep Subsection

:INITiate[:IMMediate]

:INITiate:CONTinuous

ABORt

Command Format	**:INITiate[:IMMediate]**
Instruction	Restarts the current sweep. :INITiate:RESTart and :INITiate:IMMediate perform exactly the same function.

Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	
Example	:INITiate:IMMediate

Command Format	**:INITiate:CONTinuous OFF|ON|0|1** **:INITiate:CONTinuous?**
Instruction	Sets continuous sweep mode on or off. Gets continuous sweep mode state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	ON
Menu	Sweep > Sweep
Example	:INITiate:CONTinuous OFF

Command Format	**ABORt**
Instruction	This command is used to stop the current measurement. It aborts the currentmeasurement as quickly as possible, resets the sweep and trigger systems, and puts the measurement into an "idle" state. If the analyzer is set for Continuous measurement, it sets up the measurement and initiates a new data measurement sequence with a new data acquisition (sweep) taken once the trigger condition is met. If the analyzer is set for Single measurement, it remains in the "idle" state until an :INIT:IMM command is received.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	None
Example	INIT;ABORt

6.4 Trace Subsection

Command Format	**DISP:WINDow:TRACe:STATe** **DISP:WINDow:TRACe:STATe?**
Instruction	Turns on oroff trace data display state. Gets trace data display state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	ON
Menu	Trace > Display
Example	DISP:WINDow:TRACe:STATe OFF

Command Format	**DISP:WINDow:TRACe:MEMory[:STATe]** **DISP:WINDow:TRACe:MEMory[:STATe]?**
Instruction	Turns on oroff trace memory display state. Gets trace memory display state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	OFF
Menu	Trace > Display
Example	DISP:WINDow:TRACe: MEMory OFF

Command Format	**:CALCulate[:SELected]:MATH:MEMorize**
Instruction	Copies trace data to memory.
Parameter Type	None
Parameter Range	None
Return	None

Default	None
Menu	Trace > Data→Mem
Example	:CALCulate:SELected:MATH:MEMorize

Command Format	**[:SENSe]:AVERage:COUNt <integer>** **[:SENSe]:AVERage:COUNt?**
Instruction	Sets trace average count. Gets trace average count.
Parameter Type	Integer
Parameter Range	1 ~ 999
Return	Interger
Default	100
Menu	Trace > Average
Example	:AVERage:TRACe:COUNt 200

Command Format	**[:SENSe]:AVERage:STATe OFF|ON|0|1** **[:SENSe]:AVERage:STATe?**
Instruction	Sets trace average states. Gets trace average states.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	OFF
Menu	Trace > Average
Example	:AVERage:TRACe:STATe ON

6.5 Marker Subsection

:CALCulate:MARKer[1]|2|3|4|:STATe

:CALCulate:MARKer[1]|2|3|4:MODE

:CALCulate:MARKer[1]|2|3|4:X

:CALCulate:MARKer[1]|2|3|4:Y?

:CALCulate:MARKer[1]|2|3|4:CPSearch[:STATe]

:CALCulate:MARKer[1]|2|3|4:CVSearch[:STATe]

:CALCulate[:SELected]:MARKer:AOFF

:CALCulate:MARKer[1]|2|3|4:MAXimum

:CALCulate:MARKer[1]|2|3|4:MINimize

Command Format	**:CALCulate:MARKer[1]|2|3|4|:STATe OFF|ON|0|1** **:CALCulate:MARKer[1]|2|3|4|:STATe?**
Instruction	Sets marker state Gets marker state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	OFF
Menu	Marker
Example	:CALCulate:MARK1:STATe ON

Command Format	**:CALCulate:MARKer[1]|2|3|4:MODE POSition|DELTa|OFF** **:CALCulate:MARKer[1]|2|3|4:MODE?**
Instruction	Sets marker mode. Gets marker mode.
Parameter Type	Enumeration
Parameter Range	POSition DELTa OFF
Return	Enumeration: POS|DELT|OFF
Default	OFF
Menu	Marker
Example	:CALCulate:MARK1:MODE POSition

Command Format	**:CALCulate:MARKer[1]|2|3|4:X <para>** **:CALCulate:MARKer[1]|2|3|4:X?**
Instruction	Sets marker X value. Gets marker X value. This command only works when marker is not off.
Parameter Type	Float, Unit: Hz, kHz, MHz, GHz， m
Parameter Range	When meas type is CAT: 100 kHz ~ 3.2 GHz When meas type is DTF: 0 m ~ 34 m
Return	Float
Default	7.12m/1.59995GHz
Menu	Marker > Normal
Example	:CALCulate:MARKer4:X 0.4 m :CALCulate:MARKer4:X?

Command Format	**:CALCulate:MARKer[1]|2|3|4:Y?**
Instruction	Gets marker Y value.
Parameter Type	None
Parameter Range	None
Return	Float
Default	None
Menu	Marker > Normal
Example	:CALCulate:MARKer:Y?

Command Format	**:CALCulate:MARKer[1]|2|3|4:CPSearch[:STATe] OFF|ON|0|1** **:CALCulate:MARKer[1]|2|3|4:CPSearch[:STATe]?**
Instruction	Toggles the continuous peak search function between on and off. Gets the continuous peak search function state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	None
Menu	Marker > Cont Peak
Example	:CALCulate:MARKer1:CPSEarch ON

Command Format	**:CALCulate:MARKer[1]|2|3|4:CVSearch[:STATe] OFF|ON|0|1** **:CALCulate:MARKer[1]|2|3|4:CVSearch[:STATe]?**
Instruction	Toggles the continuous valley search function between on and off. Gets the continuous valley search function state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	None
Menu	Marker > Cont Valley
Example	:CALCulate:MARKer1:CVSEarch ON

Command Format	**:CALCulate[:SELected]:MARKer:AOFF**
Instruction	Close All Markers of current trace.
Parameter Type	None

Parameter Range	None
Return	None
Default	None
Menu	
Example	:CALCulate:MARKer:AOFF

Command Format	**:CALCulate:MARKer[1]|2|3|4:MAXimum**
Instruction	Performs a peak search.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Peak > Peak
Example	:CALCulate:MARKer4:MAXimum

Command Format	**:CALCulate:MARKer[1]|2|3|4:MINimize**
Instruction	Performs a valley search.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Peak > Valley
Example	:CALCulate:MARKer4:MINimize

6.6 Measurement Subsystem

:CALCulate:PARameter:DEFine

:CALCulate:TRANsform:DISTance:STARt

:CALCulate:TRANsform:DISTance:STOP

:CALCulate:TRANsform:DISTance:UNIT

:CORRection:RVELocity:COAX

:CORRection:LOSS:COAX

:CALCulate:TRANsform:DISTance:WINDow

[:SENSe]:CORRection:COLLect:METHod:SOLT1

[:SENSe]:CORRection:COLLect[:ACQuire]:LOAD

[:SENSe]:CORRection:COLLect[:ACQuire]:OPEN

[:SENSe]:CORRection:COLLect[:ACQuire]:SHORt

[:SENSe]:CORRection:COLLect:METHod:TYPE?

[:SENSe]:CORRection:COLLect:CLEar

[:SENSe]:CORRection:COLLect:SAVE

Command Format	**:CALCulate:PARameter:DEFine DTF|CAT** **:CALCulate:PARameter:DEFine?**
Instruction	Sets meas type. Gets meas type.
Parameter Type	Enumeration
Parameter Range	DTF CAT
Return	Enumeration: DTF|CAT
Default	REFLcoe
Menu	Meas
Example	:CALCulate:PARameter:DEFine DTF

Command Format	**:CALCulate:TRANsform:DISTance:STARt <value>** **:CALCulate:TRANsform:DISTance:STARt?**
Instruction	Sets Start Distance. Gets Start Distance.
Parameter Type	Float, Unit: m
Parameter Range	0 m ~ 33.8 m
Return	Float
Default	0 m
Menu	Meas > Start Distance
Example	:CALCulate:TRANsform:DISTance:STARt 0.5 m

Command Format	**:CALCulate:TRANsform:DISTance:STOP <value>** **:CALCulate:TRANsform:DISTance:STOP?**
Instruction	Sets Start Distance. Gets Start Distance.
Parameter Type	Float, Unit: m
Parameter Range	0.2 m ~ 34 m

Return	Float
Default	30.6 m
Menu	Meas > Stop Distance
Example	:CALCulate:TRANsform:DISTance:STOP 32.5 m

Command Format	**:CALCulate:TRANsform:DISTance:UNIT METers|FEET** **:CALCulate:TRANsform:DISTance:UNIT?**
Instruction	Sets unit. Gets unit.
Parameter Type	Enumeration
Parameter Range	METers FEET
Return	Enumeration: MET|FEET
Default	MET
Menu	Meas > Unit
Example	:CALCulate:TRANsform:DISTance:UNIT FEET

Command Format	**:CORRection:RVELocity:COAX <value>** **:CORRection:RVELocity:COAX?**
Instruction	Sets velocity factor. Gets velocity factor.
Parameter Type	Float
Parameter Range	10 ~ 100
Return	Float
Default	66
Menu	Meas > Velocity Factor
Example	:CORRection:RVELocity:COAX 12.34

Command Format	**:CORRection:LOSS:COAX <value>** **:CORRection:LOSS:COAX?**
Instruction	Sets cable atten. Gets cable atten.
Parameter Type	Float
Parameter Range	-1000 ~ 10000
Return	Float
Default	0
Menu	Meas > Cable Atten

Example	:CORRection:LOSS:COAX 12.34

Command Format	**:CALCulate:TRANsform:DISTance:WINDow OFF|RECT|HAMM** **:CALCulate:TRANsform:DISTance:WINDow?**
Instruction	Sets window. Gets window.
Parameter Type	Enumeration
Parameter Range	OFF RECT HAMM
Return	Enumeration: OFF|RECT|HAMM
Default	HAMM
Menu	Meas > Window
Example	:CALCulate:TRANsform:DISTance:WINDow RECT

Command Format	**[:SENSe]:CORRection:COLLect:METHod:SOLT1**
Instruction	Sets the Cal Method to 1-port SOLT calibration.
Parameter Type	Integer
Parameter Range	1
Return	None
Default	None
Menu	Meas > Cailbration > Calibrate
Example	:CORRection:COLLect:METHod:SOLT1 1

Command Format	**[:SENSe]:CORRection:COLLect[:ACQuire]:LOAD**
Instruction	Measures the Load calibration standard that is connected to the specified port.
Parameter Type	Integer
Parameter Range	1
Return	None
Default	1
Menu	
Example	:CORRection:COLLect:LOAD 1

Command Format	[:SENSe]:CORRection:COLLect[:ACQuire]:OPEN
Instruction	Measures the Open calibration standard that is connected to the specified port.
Parameter Type	Integer
Parameter Range	1
Return	None
Default	1
Menu	
Example	:CORRection:COLLect:OPEN 1

Command Format	[:SENSe]:CORRection:COLLect[:ACQuire]:SHORt
Instruction	Measures the Short calibration standard that is connected to the specified port.
Parameter Type	Integer
Parameter Range	1
Return	None
Default	1
Menu	
Example	:CORRection:COLLect:SHOR 1

Command Format	[:SENSe]:CORRection:COLLect:METHod:TYPE?
Instruction	Querys Calibration type.
Parameter Type	None
Parameter Range	None
Return	"NONE", " SOLT1", ” RESPT”
Default	
Menu	
Example	:CORRection:COLLect:METHod:TYPE?

Command Format	[:SENSe]:CORRection:COLLect:CLEar
Instruction	Clears Calibration Data.
Parameter Type	None

Parameter Range	None
Return	None
Default	None
Menu	
Example	:CORRection:COLLect:CLEar

Command Format	**[:SENSe]:CORRection:COLLect:SAVE**
Instruction	Ends the calibration.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	
Example	:CORRection:COLLect:SAVE

7.Modulation Analyzer

7.1 Frequency Subsection

[:SENSe]:FREQuency:CENTer

[:SENSe]:FREQuency:CENTer:STEP[:INCRement]

[:SENSe]:FREQuency:SPAN?

Command Format	[:SENSe]:FREQuency:CENTer <freq> [:SENSe]:FREQuency:CENTer?
Instruction	Sets the center frequency. Gets the center frequency.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	0 Hz ~ 3.2 GHz
Return	Float, unit: Hz
Default	100 MHz
Menu	Frequency > Center Freq
Example	[:SENSe]:FREQuency:CENTer 300 MHz

Command Format	[:SENSe]:FREQuency:CENTer:STEP[:INCRement] <freq> [:SENSe]:FREQuency:CENTer:STEP[:INCRement]?
Instruction	Sets frequency step. Gets frequency step.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	1 Hz ~ 100 MHz

Return	Float, unit: Hz
Default	10 kHz
Menu	Frequency > Freq Step
Example	[:SENSe]:FREQuency:CENTer:STEP[:INCRement] 20 MHz

Command Format	**[:SENSe]:FREQuency:SPAN?**
Instruction	Querys span. The span of modulation analyzer mode is determined by multiple measurement parameters, and can not be set directly.
Parameter Type	None
Parameter Range	None
Return	Float, unit: Hz
Default	31.25 kHz
Menu	Span > Span
Example	[:SENSe]:FREQuency:SPAN?

7.2 Amplitude Subsection

[:SENSe]:POWer[:RF]:ATTenuation

[:SENSe]:POWer[:RF]:ATTenuation:AUTO

:TRACe1|2|3|4:Y[:SCALe]:RLEVel

:TRACe1|2|3|4:Y[:SCALe]:PDIVision

:TRACe1|2|3|4[:Y]:AUToscale

Command Format	**[:SENSe]:POWer[:RF]:ATTenuation <value>** **[:SENSe]:POWer[:RF]:ATTenuation?**
Instruction	Sets the input attenuator. Gets the input attenuator.
Parameter Type	Integer, unit: dB
Parameter Range	0 dB ~ 51 dB
Return	Integer, unit: dB
Default	20 dB
Menu	Amplitude > Attenuator
Example	[:SENSe]:POWer[:RF]:ATTenuation 30 dB

Command Format	**[:SENSe]:POWer[:RF]:ATTenuation:AUTO OFF|ON|0|1** **[:SENSe]:POWer[:RF]:ATTenuation:AUTO?**
Instruction	Sets the input attenuator. Gets the input attenuator.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	0
Menu	Amplitude > Attenuator
Example	[:SENSe]:POWer[:RF]:ATTenuation:AUTO ON

Command Format	**:TRACe1|2|3|4:Y[:SCALe]:RLEVel <value>** **:TRACe1|2|3|4:Y[:SCALe]:RLEVel?**
Instruction	This command sets the reference level for the Y-axis. Gets reference level. The command is valid if the measurement mode is ASK, FSK, MSK, PSK, QAM and the data format is not Syms/Errs.
Parameter Type	Float
Parameter Range	If the display type is Log Mag: -1000 ~ 1000 If the display type is Lin Mag: -1000 ~ 1000 If the display type is Real: -1000 ~ 1000 If the display type is Imag: -1000 ~ 1000 If the display type is I-Q: -1000 ~ 1000 If the display type is Constellation: -1000 ~ 1000 If the display type is I-Eye: -1000 ~ 1000 If the display type is Q-Eye: -1000 ~ 1000 If the display type is Wrap Phase: -1000 ~ 1000 If the display type is Unwrap Phase: -1000 ~ 1000 If the display type is Trellis-Eye: -1e5 ~ 1e9
Return	Float
Default	
Menu	Amplitude > Ref Level
Example	:TRACe4:Y:RLEVel 2

Command Format	**:TRACe1|2|3|4:Y[:SCALe]:PDIVision <value>** **:TRACe1|2|3|4:Y[:SCALe]:PDIVision?**
Instruction	This command sets the per-division display scaling for the y-axis. Gets Scale/Div when scale type. The command is valid if the measurement mode is ASK, FSK, MSK, PSK, QAM and the data format is not Syms/Errs.
Parameter Type	Float
Parameter Range	
Return	Float
Default	

Menu	Amplitude > Scale/Div
Example	:TRACe4:Y:PDIVision 2

Command Format	**:TRACe1|2|3|4[:Y]:AUToscale**
Instruction	Sets auto scale.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Amplitude > Auto Scale
Example	:TRACe2:AUToscale

7.3 BW Subsection

[:SENSe]:BWIDth[:RESolution]?

[:SENSe]:DDEMod:FFT:WINDow:TYPE

Command Format	**[:SENSe]:BWIDth[:RESolution]?**
Instruction	Querys equalization BW.
Parameter Type	None
Parameter Range	None
Return	Float, unit: Hz
Default	100 kHz
Menu	BW > EQBW
Example	:BWIDth?

Command Format	**[:SENSe]:DDEMod:FFT:WINDow:TYPE** **[:SENSe]:DDEMod:FFT:WINDow:TYPE?**
Instruction	Sets FFT window function. Gets FFT window function.
Parameter Type	Enumeration RECTangular HAMMing : HANNing FLATtop BLACkman

Parameter Range	None
Return	Enumeration RECT HAMM HANN FLAT BLAC
Default	100 kHz
Menu	BW > Window
Example	:DDEMod:FFT:WINDow:TYPE BLAC

7.4 Sweep Subsection

:INITiat[:IMMediate]

:INITiate:CONTinuous

ABORt

Command Format	**:INITiat[:IMMediate]**
Instruction	Restart the current sweep. :INITiate:RESTart and :INITiate:IMMediate perform exactly the same function.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	
Example	:INITiate:IMMediate

Command Format	**:INITiate:CONTinuous OFF|ON|0|1** **:INITiate:CONTinuous?**
Instruction	Sets continuous sweep mode on-off. Gets continuous sweep mode state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	ON
Menu	Sweep > Sweep
Example	:INITiate:CONTinuous OFF

Command Format	ABORt
Instruction	This command is used to stop the current measurement. It aborts the currentmeasurement as quickly as possible, resets the sweep and trigger systems, and puts the measurement into an "idle" state. If the analyzer is set for Continuous measurement, it sets up the measurement and initiates a new data measurement sequence with a new data acquisition (sweep) taken once the trigger condition is met. If the analyzer is set for Single measurement, it remains in the "idle" state until an :INIT:IMM command is received.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	None
Example	INIT;ABORt

7.5 Trace Subsection

:CALCulate:PARameter:COUNt

:DISPlay:LAYout

:TRACe[1]|2|3|4:DATA:NAME

:TRACe[1]|2|3|4:FORMat[:Y]

:TRACe:COPY

:TRACe:DEMod:EYE:LENGth

:TRACe:DEMod:TABLe:FORMat

Command Format	:CALCulate:PARameter:COUNt <integer> :CALCulate:PARameter:COUNt?
Instruction	Sets trace number. Gets trace number.
Parameter Type	Integer
Parameter Range	1 ~ 4
Return	Integer
Default	1
Menu	Trace > Num of Traces
Example	:CALCulate:PARameter:COUNt 4

Command Format	**:DISPlay:LAYout <integer,integer>**
Instruction	Sets trace layout on screen. Currently, one row, two columns are not supported (1, 2)
Parameter Type	Integer (rows, columns)
Parameter Range	rows 1 ~ 2 columns 1 ~ 2
Return	
Default	two rows, two columns
Menu	Trace > Layout
Example	:DISPlay:LAYout 2,2

Command Format	**:TRACe[1]|2|3|4:DATA:NAME** **:TRACe[1]|2|3|4:DATA:NAME?**
Instruction	Sets trace format. Gets trace format.
Parameter Type	Enumeration
Parameter Range	TIME: time SPECtrum: spectrum MTIMe: IQ meas time MSPEctrum: IQ meas spectrum (FFT of IQ Meas Time.) RTIMe: IQ Reference time (Reconstructed ideal time waveform to compare IQ Meas Time against) RSPEctrum: IQ Reference spectrum (FFT of IQ Reference time.) MERRor: IQ Mag Err (Difference in length of the IQ Meas Time vector and IQ Ref Time vectorat each point in time.) PERRor: IQ Phase Err (Difference in phase of the IQ Meas Time vector and IQ Ref Time vector at each point in time.) EVTime: Error Time (Vector difference between IQ Meas Time and IQ Ref Time at each pointin time.) EVSPectrum: Error Vector Spec SYMSerrs: Syms/Errs
Return	Enumeration
Default	
Menu	Trace > Format
Example	:TRACe:DATA:NAME SYMS

Command Format	**:TRACe[1]|2|3|4:FORMat[:Y]** **:TRACe[1]|2|3|4:FORMat[:Y]?**
Instruction	Sets trace format Gets trace format
Parameter Type	Enumeration
Parameter Range	MLOG: Log Mag MLINear: Lin Mag REAL: Real IMAGinary: Imag IQ: I-Q CONStln: Constellation IEYE: I-Eye QEYE: Q-Eye WPHAse: Wrap Phase UWPHase: Unwrap Phase TRELlis: Trellis-Eye
Return	MLOG MLIN REAL IMAG IQ CONS IEYE QEYE WPHA UWPH TREL
Default	
Menu	Trace > Format
Example	:TRACe:FORMat MLIN

Command Format	**:TRACe:COPY <from,to>**
Instruction	Copies trace data to another trace.
Parameter Type	Enumeration
Parameter Range	A|B|C|D orTRACE1| TRACE2| TRACE2| TRACE4
Return	None
Default	None
Menu	Trace > Copy To
Example	:TRACe:COPY A,B :TRACe:COPY TRACE1,TRACE2

Command Format	**:TRACe:DEMod:EYE:LENGth <integer>** **:TRACe:DEMod:EYE:LENGth?**
Instruction	Sets eye length. Gets eye length.
Parameter	Integer

Type	
Parameter Range	2 ~ 40
Return	Integer
Default	2
Menu	Trace > Properties
Example	:TRACe:DEMod:EYE:LENGth 4

Command Format	**:TRACe:DEMod:TABLe:FORMat** **:TRACe:DEMod:TABLe:FORMat?**
Instruction	Displays format of Symbol Table data.
Parameter Type	Enumeration
Parameter Range	BINary|HEXadecimal
Return	Enumeration BIN|HEX
Default	HEX
Menu	Trace > Properties
Example	:TRACe:DEMod:TABLe:FORMat HEX

7.6 Marker Subsection

:TRACe[1]|2|3|4:MARKer[1]|2|3|4:ENABle

:TRACe[1]|2|3|4:MARKer[1]|2|3|4:TYPE

:TRACe[1]|2|3|4:MARKer[1]|2|3|4:X

:TRACe[1]|2|3|4:MARKer[1]|2|3|4:Y?

:TRACe[1]|2|3|4:MARKer[1]|2|3|4:REFerence

:CALCulate[:SELected]:MARKer:COUPle

:CALCulate:MARKer:AOFF

Command Format	**:TRACe[1]|2|3|4:MARKer[1]|2|3|4:ENABle OFF|ON|0|1** **:TRACe[1]|2|3|4:MARKer[1]|2|3|4:ENABle?**
Instruction	Sets marker state. Gets marker state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	OFF
Menu	Marker

Example	:TRACe1:MARKer1:ENABle ON

Command Format	**:TRACe[1]|2|3|4:MARKer[1]|2|3|4:TYPE POSition|DELTa|OFF** **:TRACe[1]|2|3|4:MARKer[1]|2|3|4:TYPE?**
Instruction	Sets marker mode. Gets marker mode.
Parameter Type	Enumeration
Parameter Range	POSition DELTa OFF
Return	Enumeration: POS|DELT|OFF
Default	OFF
Menu	Marker
Example	:TRACe:MARKer:TYPE POSition

Command Format	**:TRACe[1]|2|3|4:MARKer[1]|2|3|4:X <para>** **:TRACe[1]|2|3|4:MARKer[1]|2|3|4:X?**
Instruction	Sets marker X value. Gets marker X value. This command only works when marker is not off.
Parameter Type	Float
Parameter Range	
Return	Float
Default	
Menu	Marker > Normal
Example	:TRACe:MARKer:X 200 :TRACe:MARKer:X?

Command Format	**:TRACe[1]|2|3|4:MARKer[1]|2|3|4:Y?**
Instruction	Gets marker Y value.
Parameter Type	None
Parameter Range	None
Return	Float
Default	None
Menu	None
Example	:TRACe:MARKer:Y?

Command Format	**:TRACe[1]|2|3|4:MARKer[1]|2|3|4:REFerence <integer>** **:TRACe[1]|2|3|4:MARKer[1]|2|3|4:REFerence?**
Instruction	Sets reference marker. Gets reference marker. Cannot set the current marker to the reference marker.
Parameter Type	Integer
Parameter Range	1 ~ 4
Return	1 ~ 4
Default	2
Menu	Marker > Relative To
Example	:TRACe:MARKer:REFerence 3

Command Format	**:CALCulate[:SELected]:MARKer:COUPle OFF|ON|0|1** **:CALCulate[:SELected]:MARKer:COUPle?**
Instruction	Sets marker couple state. Gets marker couple state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	None
Menu	Marker > Couple
Example	:CALCulate:MARKer:COUPle ON

Command Format	**:CALCulate:MARKer:AOFF**
Instruction	Close all markers.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Marker > All Off
Example	:CALCulate:MARKer:AOFF

7.7 Measurement Subsystem

[:SENSe]:DDEMod:MODulation

[:SENSe]:ADEMod:STYLe

:DDEMod[:FORMat]:SRATe

[:SENSe]:DDEMod[:FORMat]:SYMBol:POINts

[:SENSe]:DDEMod[:FORMat]:RLENgth

[:SENSe]:DDEMod:FILTer[:MEASurement]

[:SENSe]:DDEMod:FILTer:REFerence

[:SENSe]:STATistic:STATe

[:SENSe]:AVERage[:STATe]

[:SENSe]:AVERage:COUNt

:CALCulate:RESTart

:READ:DDEMod?

Command Format	**[:SENSe]:DDEMod:MODulation** **[:SENSe]:DDEMod:MODulation?**
Instruction	Sets Digital Demodulation Mode. Gets Digital Demodulation Mode.
Parameter Type	Enumeration
Parameter Range	ASK2 MSK BPSK QPSK PSK8 DBPSK DQPSK DPSK8 OQPSK PI4DQ PI8D8 QAM16 QAM32 QAM64 QAM128 QAM256 FSK2 FSK4 FSK8 FSK16
Return	Enumeration
Default	QAM16
Menu	Meas
Example	:DDEMod:MODulation FSK8

Command Format	**[:SENSe]:ADEMod:STYLe** **[:SENSe]:ADEMod:STYLe?**
Instruction	Sets Analog Modulation Type. Gets Analog Modulation Type.

Parameter Type	Enumeration
Parameter Range	AM FM
Return	Enumeration: AM|FM
Default	AM
Menu	Meas
Example	:ADEMod:STYLe AM

Command Format	**:DDEMod[:FORMat]:SRATe <integer>** **:DDEMod[:FORMat]:SRATe?**
Instruction	Sets Symbol Rate. Gets Symbol Rate.
Parameter Type	Integer
Parameter Range	1000 ~ 2500000
Return	Integer
Default	10000
Menu	Meas > Symbol Rate
Example	:DDEMod:SRATe 2000

Command Format	**[:SENSe]:DDEMod[:FORMat]:SYMBol:POINts <integer>** **[:SENSe]:DDEMod[:FORMat]:SYMBol:POINts?**
Instruction	Sets Points per Symbol. Gets Points per Symbol.
Parameter Type	Discrete
Parameter Range	4, 6, 8, 10, 12, 14, 16
Return	Discrete
Default	4
Menu	Meas > Points/Symbol
Example	DDEMod:SYMBol:POINts 14

Command Format	**[:SENSe]:DDEMod[:FORMat]:RLENgth <integer>** **[:SENSe]:DDEMod[:FORMat]:RLENgth?**
Instruction	Sets meas length. Gets meas length.
Parameter Type	Integer
Parameter Range	16 ~ 4096
Return	Integer

Default	128
Menu	Meas > Meas Length
Example	:DDEMod:RLENgth 200

Command Format	**[:SENSe]:DDEMod:FILTer[:MEASurement]** **[:SENSe]:DDEMod:FILTer[:MEASurement]?**
Instruction	Sets meas filter. Gets meas filter.
Parameter Type	Enumeration
Parameter Range	OFF RRCosine RCOSine GAUSsian HSIN
Return	0|1
Default	ASK, FSK, PSK, QAM Default is RCOSine MSK Default is OFF
Menu	Meas > Filter Setup > Meas Filter
Example	:DDEMod:FILTer HSIN

Command Format	**[:SENSe]:DDEMod:FILTer:REFerence** **[:SENSe]:DDEMod:FILTer:REFerence?**
Instruction	Sets reference filter. Gets reference filter.
Parameter Type	Enumeration
Parameter Range	OFF RRCosine: Root Raised Cosine RCOSine : Raised Cosine GAUSsian HSIN：Half Sine
Return	Enumeration
Default	ASK, FSK, PSK, QAM Default is RRC MSK Default is GAUS
Menu	Meas > Ref Filter
Example	:DDEMod:FILTer:REFerence OFF

Command Format	**[:SENSe]:STATistic:STATe** **[:SENSe]:STATistic:STATe?**
Instruction	Sets Meas Statistic State. Gets Meas Statistic State.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1

Return	Enumeration
Default	OFF
Menu	Meas > Statistic > Statistic
Example	:STATistic:STATe ON

Command Format	**[:SENSe]:AVERage[:STATe]** **[:SENSe]:AVERage[:STATe]?**
Instruction	Sets meas average state. Gets meas average state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	Boolean
Default	OFF
Menu	Meas> Statistic > Avg
Example	:AVERage ON

Command Format	**[:SENSe]:AVERage:COUNt** **[:SENSe]:AVERage:COUNt?**
Instruction	Sets meas average count. Gets meas average count.
Parameter Type	Integer
Parameter Range	1 ~ 1000
Return	Integer
Default	10
Menu	Meas> Statistic > Avg
Example	:AVERage:COUNt 20

Command Format	**:CALCulate:RESTart**
Instruction	Restarts measurements.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Meas > Statistic > Restart Meas

Example	:CALCulate:RESTart

Command Format	**:READ:DDEMod?**
Instruction	Read digital demod result. If demod type is ASK it will return: 1. ASK err rms (% rms) 2. ASK err peak (% pk) 3. symbol position of ASK err peak 4. carrier power 5. carrier offset 6. ASK depth If demod type is FSK it will return: 1. FSK err rms (% rms) 2. FSK err peak (% pk) 3. symbol position of FSK err peak 4. carrier power 5. carrier offset 6. FSK deviation If If demod type is MSK,PSK,QAM it will return: 1. EVM rms (% rms) 2. EVM peak (% pk) 3. symbol position of EVM peak 4. magnitude error rms (% rms). 5. magnitude error peak (% pk) 6. symbol position of magnitude error peak 7. phase error rms (deg) 8. phase error peak (deg pk) 9. symbol position of phase error peak 10. frequency error (Hz) 11. IQ offset 12. SNR(MER) (dB) 13. quadrature error (deg) 14. gain imbalance (dB)
Parameter Type	None
Parameter Range	None
Return	String
Default	None
Menu	
Example	:READ:DDEMod?

7.8 Trigger Subsection

:TRIGger[:SEQuence]:SOURce

:TRIGger[:SEQuence]:RF:LEVel

:TRIGger[:SEQuence]:RFBurst:SLOPe

Command Format	**:TRIGger[:SEQuence]:SOURce IMMediate| RFBurst |EXTernal** **:TRIGger[:SEQuence]:SOURce?**
Instruction	Specifies the source (or type) of triggering used to start a measurement. Gets trigger type. RFBurst is not supported if demod type is MSK, PSK, QAM.
Parameter Type	Enumeration
Parameter Range	IMMediate: free-run triggering. RFBurst: triggers on the RF signal level. EXTernal: allows you to connect an external trigger source.
Return	Enumeration: IMM|EXT|RFB
Default	IMMediate
Menu	Trigger
Example	:TRIGger:SOURce IMMediate

Command Format	**:TRIGger[:SEQuence]:RF:LEVel <value>** **:TRIGger[:SEQuence]:RF:LEVel?**
Instruction	Sets RF Trigger Level. Gets RF Trigger Level.
Parameter Type	Float, Unit: dBm
Parameter Range	-300 dBm ~ 50 dBm
Return	Float, Unit: dBm
Default	0 dBm
Menu	Trigger > RF Trigger
Example	:TRIGger:RF:LEVel 0.5 dBm

Command Format	**:TRIGger[:SEQuence]:RFBurst:SLOPe POSitive|NEGative** **:TRIGger[:SEQuence]:RFBurst:SLOPe?**
Instruction	Sets trigger edge. Gets trigger edge.
Parameter Type	Enumeration
Parameter Range	POSitive NEGative
Return	Enumeration: POS|NEG
Default	POSitive
Menu	Trigger > External
Example	:TRIGger:RFBurst:SLOPe POSitive

8.Real Time Spectrum Analysis

The model supported by real-time spectrum analysis is ssa3000X-R series

8.1 Frequency Subsection

[:SENSe]:FREQuency:CENTer

[:SENSe]:FREQuency:CENTer:STEP[:INCRement]

[:SENSe]:FREQuency:STARt <freq>

[:SENSe]:FREQuency:STOP <freq>

[:SENSe]:FREQuency:CENTer:STEP:AUTO

[:SENSe]:FREQuency:OFFSet

[:SENSe]:FREQuency:SPAN

[:SENSe]:FREQuency:SPAN:FULL

[:SENSe]:FREQuency:SPAN:ZERO

[:SENSe]:FREQuency:SPAN:PREVious

[:SENSe]:FREQuency:SPAN:HALF

[:SENSe]:FREQuency:SPAN:DOUBle

Command Format	**[:SENSe]:FREQuency:CENTer <freq>** **[:SENSe]:FREQuency:CENTer?**
Instruction	Sets the center frequency. Gets the center frequency.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	2.5 kHz ~ 7.4999975 GHz

Return	Float, unit: Hz
Default	20 MHz
Menu	Frequency > Center Freq
Example	[:SENSe]:FREQuency:CENTer 300 MHz

Command Format	**[:SENSe]:FREQuency:CENTer:STEP[:INCRement] <freq>** **[:SENSe]:FREQuency:CENTer:STEP[:INCRement]?**
Instruction	Sets frequency step. Gets frequency step.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	1 Hz ~ 999.999 kHz
Return	Float, unit: Hz
Default	320 MHz
Menu	Frequency > Freq Step
Example	[:SENSe]:FREQuency:CENTer:STEP[:INCRement] 20 MHz

Command Format	**[:SENSe]:FREQuency:STARt <freq>** **[:SENSe]:FREQuency:STARt?**
Instruction	Sets the start frequency. Gets the start Frequency.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	0 Hz ~ 7.499995 GHz
Return	Float, unit: Hz
Default	0 Hz
Menu	Frequency > Start Freq
Example	:FREQuency:STARt 100 Hz

Command Format	**[:SENSe]:FREQuency:STOP <freq>** **[:SENSe]:FREQuency:STOP?**
Instruction	Sets the stop frequency. Gets the stop frequency.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	5 kHz ~ 7.5 GHz
Return	Float, unit: Hz
Default	40 MHz
Menu	Frequency > Stop Freq

Example	:FREQuency:STOP 1.0 GHz

Command Format	**[:SENSe]:FREQuency:CENTer:STEP:AUTO OFF|ON|0|1** **[:SENSe]:FREQuency:CENTer:STEP:AUTO?**
Instruction	Specifies whether the step size is set automatically based on the span. Gets center frequency step mode.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	ON
Menu	Frequency > Freq Step
Example	:FREQuency:CENTer:STEP:AUTO OFF

Command Format	**[:SENSe]:FREQuency:OFFSet <freq>** **[:SENSe]:FREQuency:OFFSet?**
Instruction	Sets the frequency offset. Gets the frequency offset.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	-100 GHz ~ 100 GHz
Return	Float, unit: Hz
Default	0 Hz
Menu	Frequency > Freq Offset
Example	:FREQuency:OFFSet 1 GHz

Command Format	**[:SENSe]:FREQuency:SPAN <freq>** **[:SENSe]:FREQuency:SPAN?**
Instruction	Sets the span frequency. Gets the span frequency..
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	5 kHz ~ 40 MHz
Return	Float, unit: Hz
Default	40 MHz
Menu	Span > Span
Example	:FREQuency:SPAN 1 GHz

Command Format	**[:SENSe]:FREQuency:SPAN:FULL**
Instruction	Sets the frequency span to full scale.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Span > Full Span
Example	:FREQuency:SPAN:FULL

Command Format	**[:SENSe]:FREQuency:SPAN:ZERO**
Instruction	Sets the frequency span to zero span.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Span > Zero Span
Example	:FREQuency:SPAN:ZERO

Command Format	**[:SENSe]:FREQuency:SPAN:PREVious**
Instruction	Sets the frequency span to the previous span setting.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Span > Last Span
Example	:FREQuency:SPAN:PREVious

Command Format	**[:SENSe]:FREQuency:SPAN:HALF**
Instruction	Sets the frequency span to half of the current span setting.
Parameter Type	None

Parameter Range	None
Return	None
Default	None
Menu	Span> Zoom In
Example	:FREQuency:SPAN:HALF

Command Format	**[:SENSe]:FREQuency:SPAN:DOUBle**
Instruction	Sets the frequency span to double the current span setting.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Span> Zoom Out
Example	:FREQuency:SPAN:DOUBle

8.2 Amplitude Subsection

[:SENSe]:POWer[:RF]:ATTenuation

[:SENSe]:POWer[:RF]:ATTenuation:AUTO

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

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

[:SENSe]:POWer[:RF]:GAIN[:STATe] :UNIT:POWer

Command Format	**[:SENSe]:POWer[:RF]:ATTenuation <value>** **[:SENSe]:POWer[:RF]:ATTenuation?**
Instruction	Sets the input attenuator. Gets the input attenuator.
Parameter Type	Integer, Unit: dB
Parameter Range	0 dB ~ 50 dB
Return	Integer, unit: dB
Default	20 dB
Menu	Amplitude > Attenuator
Example	[:SENSe]:POWer[:RF]:ATTenuation 30 dB

Command Format	**[:SENSe]:POWer[:RF]:ATTenuation:AUTO OFF|ON|0|1** **[:SENSe]:POWer[:RF]:ATTenuation:AUTO?**
Instruction	Sets the input attenuator. Gets the input attenuator.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	0
Menu	Amplitude > Attenuator
Example	[:SENSe]:POWer[:RF]:ATTenuation:AUTO ON

Command Format	**:DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel <value>** **:DISPlay:WINDow:TRACe:Y[:SCALe]:RLEVel?**
Instruction	This command sets the reference level for the Y-axis. Gets reference level.
Parameter Type	Float
Parameter Range	-100 dBm ~ 30 dBm
Return	Float
Default	0 dBm
Menu	Amplitude > Ref Level
Example	:DISPlay:WINDow:TRACe:Y:RLEVel 20 DBM

Command Format	**:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision <integer>** **:DISPlay:WINDow:TRACe:Y[:SCALe]:PDIVision?**
Instruction	This command sets the per-division display scaling for the y-axis. Gets Scale/Div when scale type.
Parameter Type	Float
Parameter Range	1 dB ~ 20 dB
Return	Float
Default	10 dB
Menu	Amplitude > Scale/Div
Example	:DISPlay:WINDow:TRACe:Y:PDIVision 10 dB

Command Format	**[:SENSe]:POWer[:RF]:GAIN[:STATe] OFF|ON|0|1** **[:SENSe]:POWer[:RF]:GAIN[:STATe]?**
Instruction	Turns the internal preamp on/off. Gets preampstate.

Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	OFF
Menu	Amplitude > Preamp
Example	:POWer:GAIN ON

Command Format	**:UNIT:POWer DBM|DBMV|DBUV|V|W** **:UNIT:POWer?**
Instruction	Specifies amplitude units for the input, output and display. Gets amplitude units.
Parameter Type	Enumeration
Parameter Range	DBM|DBMV|DBUV|DBUA|V|W,
Return	Enumeration
Default	DBM
Menu	Amplitude > Units
Example	:UNIT:POWer DBMV

8.3 BW Subsection

[:SENSe]:BWIDth[:RESolution]

[:SENSe]:BWIDth[:RESolution]:AUTO

[:SENSe]:FILTer:TYPE

Command Format	**[:SENSe]:BWIDth[:RESolution] <freq>** **[:SENSe]:BWIDth[:RESolution]?**
Instruction	Specifies the resolution bandwidth. For numeric entries, all RBW types choose the nearest (arithmetically, on a linear scale, rounding up) available RBW to the value entered.
Parameter Type	None
Parameter Range	None
Return	Float, Unit: Hz
Default	1 MHz
Menu	BW > RBW
Example	:BWIDth?

Command Format	**[:SENSe]:BWIDth[:RESolution]:AUTO OFF|ON|0|1** **[:SENSe]:BWIDth[:RESolution]:AUTO?**

Instruction	Turns on/off auto resolution bandwidth state. Gets the resolution bandwidth state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	ON
Menu	BW > RBW
Example	:BWID:AUTO On

Command Format	**[:SENSe]:FILTer:TYPE** **[:SENSe]:FILTer:TYPE?**
Instruction	Sets FFT window function. Gets FFT window function.
Parameter Type	Enumeration
Parameter Range	KAISer HANNing FLATtop GAUSsian BHARris RECTangular
Return	KAIS HANN FLAT GAUS BHAR RECT
Default	100 kHz
Menu	BW > Window
Example	:FILT:TYPE KAIS

8.4 Sweep Subsection

[:SENSe]:ACQuisition:TIME

[:SENSe]:ACQuisition:TIME:AUTO

:INITiate[:IMMediate]

:INITiate:CONTinuous

:INITiate:Pause

:INITiate:RESume

Command Format	**[:SENSe]:ACQuisition:TIME** **[:SENSe]:ACQuisition:TIME？**

Instruction	Sets Acquisition time. Gets Acquisition time.
Parameter Type	Float, unit: ks, s, ms, us
Parameter Range	29.998 ms ~ 40 s
Return	Float, unit: s
Default	29.998ms
Menu	Sweep > Acq Time
Example	:ACQuisition:TIME 2s

Command Format	**[:SENSe]:ACQuisition:TIME:AUTO** **[:SENSe]:ACQuisition:TIME:AUTO?**
Instruction	This command turns on/off auto sweep time state. Gets sweep time state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	ON
Menu	Sweep > Acq Time
Example	:ACQuisition:TIME:AUTO on

Command Format	**:INITiate[:IMMediate]**
Instruction	Restarts the current sweep.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	
Example	:INITiate:IMMediate

Command Format	**:INITiate:CONTinuous OFF|ON|0|1** **:INITiate:CONTinuous?**
Instruction	Sets continuous sweep mode on-off. Gets continuous sweep mode state.
Parameter Type	Boolean
Parameter	OFF|ON|0|1

Range	
Return	0|1
Default	ON
Menu	Sweep > Sweep
Example	:INITiate:CONTinuous OFF

Command Format	**:INITiate:Pause**
Instruction	Pause current sweeep (pause at the end of the current sweep).
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Sweep > Pause
Example	:INITiate:Pause

Command Format	**:INITiate:RESume**
Instruction	Resume paused sweep
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Sweep > Resume
Example	:INITiate:RES

8.5 Trace Subsection

:TRACe[1]|2|3:MODE

:TRACe1|2|3 [:DATA]?

:TRACe[:DATA]:SPECtrum?

:TRACe[:DATA]?:PVT?

:FORMat[:TRACe][:DATA]

[:SENSe]:DETector:TRACe[1]|2|3[:FUNCtion]

[:SENSe]:DETector:TRACe:SPECtrogram

[:SENSe]:DETector:TRACe:PVTime

[:SENSe]:AVERage:TRACe[1]|2|3:COUNt

Command Format	**:TRACe[1]|2|3:MODE WRITe|MAXHold|MINHold|VIEW|BLANk|AVERage** **:TRACe[1]|2|3:MODE?**
Instruction	Selects the display mode for the selected trace.
Parameter Type	Enumeration
Parameter Range	WRITe: puts the trace in the normal mode, updating the data. MAXHold: displays the highest measured trace value for all the data that has been measured since the function was turned on. MINHold: displays the lowest measured trace value for all the data that has been measured since the function was turned on. BLANk: turns off the trace data so that it is not viewed on the display. AVERage: averages the trace for test period.
Return	Enumeration
Default	Trace1: WRITe, Trace2|3|: BLANk
Menu	Trace
Example	:TRAC1:MODE AVER

Command Format	**:TRACe1|2|3 [:DATA]?**
Instruction	This query command returns the current displayed data.
Parameter Type	None
Parameter Range	None
Return	String
Default	1
Menu	None
Example	:TRACe:DATA?

Command Format	**:TRACe[:DATA]:SPECtrum?**
Instruction	This query command returns the spectrum trace data.
Parameter Type	None
Parameter Range	None
Return	String
Default	1
Menu	None

Example	:TRACe:SPEC?

Command Format	**:TRACe[:DATA]?:PVT?**
Instruction	This query command returns the pvt trace data.
Parameter Type	None
Parameter Range	None
Return	String
Default	1
Menu	None
Example	:TRACe:PVT?

Command Format	**:FORMat[:TRACe][:DATA] ASCii|REAL** **:FORMat[:TRACe][:DATA]?**
Instruction	Sets trace data type. Gets trace data type.
Parameter Type	Enumeration
Parameter Range	ASCii REAL: single precision floating-point (float)
Return	String
Default	ASCii
Menu	None
Example	:FORMat ASCii

Command Format	**[:SENSe]:DETector:TRACe[1]|2|3[:FUNCtion]** **NEGative|POSitive|SAMPle|AVERage** **[:SENSe]:DETector:TRACe[1]|2|3[:FUNCtion]?**
Instruction	Specifies the detection mode. For each trace interval (bucket), average detection displays the average of all the samples within the interval.
Parameter Type	Enumeration
Parameter Range	NEGative: Negative peak detection displays the lowest sample taken during the interval being displayed. POSitive: Positive peak detection displays the highest sample taken during the interval being displayed. SAMPle: Sample detection displays the sample taken during the interval being displayed, and is used primarily to display noise or noise-like signals. In sample mode, the instantaneous signal value at the present display point is placed into memory. This detection should not be used to make the most accurate amplitude measurement of non noise-like signals. AVERage: Average detection is used when measuring the average value of the amplitude across each trace interval (bucket). The averaging method used by the average detector is set to either video or power as appropriate when the average type is auto coupled.

Return	Enumeration
Default	POSitive
Menu	Detect

Command Format	**[:SENSe]:DETector:TRACe:SPECtrogram** **[:SENSe]:DETector:TRACe:SPECtrogram?**
Instruction	Sets the detect type of spectrogram.
Parameter Type	Enumeration
Parameter Range	NEGative POSitive SAMPle AVERage
Return	Enumeration
Default	POS
Menu	Detect
Example	:DET:TRAC:SPEC POS

Command Format	**[:SENSe]:DETector:TRACe:PVTime** **[:SENSe]:DETector:TRACe:PVTime?**
Instruction	Sets the detect type of PVT.
Parameter Type	Enumeration
Parameter Range	NEGative POSitive SAMPle AVERage
Return	Enumeration
Default	POS
Menu	Detect
Example	:DET:TRAC:PVT POS

Command Format	**[:SENSe]:AVERage:TRACe[1]|2|3:COUNt <integer>** **[:SENSe]:AVERage:TRACe[1]|2|3:COUNt?**
Instruction	Specifies the number of measurements that are combined.
Parameter Type	Integer
Parameter Range	1 ~ 100
Return	Integer
Default	10

Menu	Trace > Average
Example	:AVERage:TRACe1:COUNt 10

8.6 Marker Subsection

:CALCulate:MARKer[1]\|2\|3\|4\|5\|6\|7\|8:MODE

:CALCulate:MARKer[1]\|2\|3\|4\|5\|6\|7\|8:TRACe

:CALCulate:MARKer[1]\|2\|3\|4\|5\|6\|7\|8:REFerence

:CALCulate:MARKer[1]\|2\|3\|4\|5\|6\|7\|8:X

:CALCulate:MARKer[1]\|2\|3\|4\|5\|6\|7\|8:Y

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8:MODE POSition|DELTa|FIXed|OFF** **:CALCulate:MARKer[1]|2|3|4|5|6|7|8:MODE?**
Instruction	Selects the type of markers that you want to activate. Gets the type of markers.
Parameter Type	Enumeration
Parameter Range	POSition DELTa FIXed OFF
Return	Enumeration
Default	OFF
Menu	Marker
Example	:CALCulate:MARK1:MODE POSition

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8:TRACe 1|2|3|4|5** **:CALCulate:MARKer[1]|2|3|4|5|6|7|8:TRACe?**
Instruction	This command assigns the specified marker to the designated trace 1, 2, 3, 4 or 5. Gets the specified marker to which trace. Trace4|5 is display trace1|2.
Parameter Type	Enumeration
Parameter Range	MARKer:1|2|3|4|5|6|7|8 TRACe:1|2|3|4|5
Return	Enumeration
Default	1
Menu	Marker > Select Trace
Example	CALCulate:MARK1:TRAC 1

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8:REFerence 1|2|3|4|5|6|7|8** **:CALCulate:MARKer[1]|2|3|4|5|6|7|8:REFerence?**
Instruction	Sets marker relative to. Gets marker relative to.
Parameter Type	Enumeration
Parameter Range	1|2|3|4|5|6|7|8
Return	Enumeration
Default	1
Menu	Marker > Relative To
Example	:CALCulate:MARKer1:REFerence 3

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8:X <para>** **:CALCulate:MARKer[1]|2|3|4|5|6|7|8:X?**
Instruction	This command positions the designated marker on its assigned trace at the specified trace X value. The value is in the X-axis units, which can be a frequency or time. The query returns the current X value of the designated marker.
Parameter Type	Frequency: Float, unit: Hz, kHz, MHz, GHz, Default “Hz” Time: Float, unit: us, ms, s, ks, Default “s”
Parameter Range	0~7.5GHz
Return	Float
Default	
Menu	Marker > Normal
Example	:CALCulate:MARKer4:X 1 kHz :CALCulate:MARKer4:X?

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6|7|8:Y** **:CALCulate:MARKer[1]|2|3|4|5|6|7|8:Y?**
Instruction	This command reads the current Y value for the designated marker. This command can be used to read the results of noise marker. Make sure that Marker is on, Reference Command: :CALCulate:MARKer[1]|2|3|4|5|6|7|8:MODE
Parameter Type	None
Parameter Range	None
Return	Float, unit: dBm
Default	None
Menu	Marker > Normal
Example	:CALCulate:MARKer1:Y?

8.7 Trigger Subsection

Command Format	**:TRIGger[:SEQuence]:SOURce IMMediate |LEVel|FMT|EXTernal** **:TRIGger[:SEQuence]:SOURce?**
Instruction	Specifies the source (or type) of triggering used to start a measurement. Gets trigger type.
Parameter Type	Enumeration
Parameter Range	IMMediate LEVel: FMT EXTernal
Return	IMM|LEV|FMT|EXT
Default	IMMediate
Menu	Trigger
Example	:TRIGger:SOURce IMMediate

Command Format	**:TRIGger[:SEQuence]:LEVel:LEVel <value>** **:TRIGger[:SEQuence]:LEVel:LEVel?**
Instruction	Specifies the level at which a pvt trigger will occur. pvt is adjusted using this command, but must also be selected using the command. Gets pvt Trigger Level.
Parameter Type	Float, unit: dBm
Parameter Range	-300 dBm ~ 50 dBm
Return	Float, unit: dBm
Default	0 dBm

Menu	Trigger > PVT > Trigger Level
Example	:TRIGger:LEVel:LEVel 0.5 dBm

Command Format	**:TRIGger[:SEQuence]:LEVel:DELay <value>** **:TRIGger[:SEQuence]:LEVel:DELay?**
Instruction	Specifies the delay at which a pvt trigger will occur. pvt is adjusted using this command, but must also be selected using the command. Gets pvt Trigger Delay.
Parameter Type	Float, unit: ks, s, ms, us, ps, ns
Parameter Range	0 ~ 25 s
Return	Float
Default	0 s
Menu	Trigger > PVT > Trigger Delay
Example	:TRIGger:LEVel:DELay 0.5 dBm

Command Format	**:TRIGger[:SEQuence]:EXTernal:DELay<value>** **:TRIGger[:SEQuence]:EXTernal:DELay?**
Instruction	Specifies the delay at which an external trigger will occur. External is adjusted using this command, but must also be selected using the command. Gets external Trigger Delay.
Parameter Type	Float, unit: ks,s,ms,us,ps,ns
Parameter Range	0~25s
Return	Float, unit: dBm
Default	0 s
Menu	Trigger > PVT > Delay
Example	:TRIGger[:SEQuence]: EXTernal: DELay 2

Command Format	**:TRIGger[:SEQuence]:EXTernal:SLOPe POSitive|NEGative** **:TRIGger[:SEQuence]:EXTernal:SLOPe?**
Instruction	Sets Trigger edge. Gets Trigger edge.
Parameter Type	Enumeration
Parameter Range	POSitive: positive edge. NEGative: negative edge.
Return	Enumeration
Default	POSitive
Menu	Trigger > External Trigger > Trigger edge
Example	:TRIGger:EXTernal:SLOPe POSitive

Command Format	**:TRIGger[:SEQuence]:FMT:TCRiteria GTLower|LTLower|OUTSide|INSide** **:TRIGger[:SEQuence]:FMT:TCRiteria?**
Instruction	Sets FMT Trigger type. Gets FMT Trigger type.
Parameter Type	Enumeration
Parameter Range	GTLower LTLower OUTSide INSide
Return	GTLower | LTLower| OUTSide| INSide
Default	GTL
Menu	Limit > Mask Edit > Mask Type
Example	:TRIGger:FMT:TCRiteria INS

Command Format	**TRIGger[:SEQuence]:FMT:MASK:BUILd**
Instruction	Auto creats a FMT MASK according to waveform shape.
Parameter Type	Enumeration
Parameter Range	None
Return	None
Default	None
Menu	Limit > Mask Edit > Build
Example	TRIGger:FMT:MASK:BUILd

Command Format	**:TRIGger[:SEQuence]:FMT:STATe** **:TRIGger[:SEQuence]:FMT:STATe?**
Instruction	Sets FMT Trigger state, Gets FMT Trigger state.
Parameter Type	Bool
Parameter Range	ON|OFF|0|1
Return	0|1
Default	OFF
Menu	Limit > FMT Enable
Example	:TRIGger:FMT:STATe on

Command Format	**:TRIGger[:SEQuence]:FMT:ACTion NORMal|BEEPer|STOP** **:TRIGger[:SEQuence]:FMT:ACTion?**
Instruction	Sets FMT Trigger action.

	Gets FMT Trigger action.
Parameter Type	Enumeration
Parameter Range	NORMal BEEPer STOP
Return	Enumeration
Default	NORM
Menu	Limit > Action
Example	:TRIGger[:SEQuence]:FMT:ACTion STOP

Command Format	**:TRIGger[:SEQuence]:EXTernal: TRACe** **:TRIGger[:SEQuence]:EXTernal: TRACe?**
Instruction	Sets FMT Trigger trace. Gets FMT Trigger trace.
Parameter Type	Interger
Parameter Range	1~3
Return	
Default	1
Menu	Limit > Mask Edit > Trace
Example	:TRIGger:EXTernal: TRACe 2

Command Format	**:TRIGger[:SEQuence]:FMT:MASK:DELete:ALL**
Instruction	Deletes all FMT mask points.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Limit > Mask Edit > Del All
Example	:TRIGger:FMT:MASK:DELete:ALL

8.8 Meas Subsection

:DISPlay:VIEW:DENSity:PERSistence:INFinite

:DISPlay:VIEW[:SELect]

:DISPlay:VIEW:DENSity:PERSistence

:DISPlay:VIEW:SPECtrogram:TRACe:NUMber

:DISPlay:VIEW:SPECtrogram:TRACe:STARt

Command Format	**:DISPlay:VIEW:DENSity:PERSistence:INFinite** **:DISPlay:VIEW:DENSity:PERSistence:INFinite?**
Instruction	Turns on or off persistence infinite mode. Querys the setting status of infinite persistence mode.
Parameter Type	Boolean
Parameter Range	ON|OFF|0|1
Return	0|1
Default	OFF
Menu	Meas Setup > Persistence
Example	:DISP:VIEW:DENS:PERS:INF ON

Command Format	**:DISPlay:VIEW[:SELect] <type>** **:DISPlay:VIEW[:SELect]?**
Instruction	Sets display type. Gets display type.
Parameter Type	Enumeration
Parameter Range	DENSity SSPectrum SPEC3D SPECtrogram PVT
Return	DENS|SSP|SPEC3D|SPEC|PVT
Default	NORM
Menu	Meas
Example	:DISP:VIEW DENS :DISP:VIEW?

Command Format	**:DISPlay:VIEW:DENSity:PERSistence** **:DISPlay:VIEW:DENSity:PERSistence?**
Instruction	Sets the Duration of persistence.
Parameter Type	Float, unit: ks,s,ms,us
Parameter Range	0 s ~ 10 s
Return	Float, unit: s
Default	0 s
Menu	Meas Setup > Persistence
Example	:DISP:VIEW:DENS:PERS 5s

Command Format	**:DISPlay:VIEW:SPECtrogram:TRACe:NUMber** **:DISPlay:VIEW:SPECtrogram:TRACe:NUMber?**
Instruction	Sets the display trace. Gets the display trace.
Parameter Type	Integer
Parameter Range	
Return	Integer
Default	1
Menu	Meas Setup > Display Trace
Example	:DISP:VIEW:SPEC:TRAC:NUM 50000 :DISP:VIEW:SPEC:TRAC:NUM?

Command Format	**:DISPlay:VIEW:SPECtrogram:TRACe:STARt** **:DISPlay:VIEW:SPECtrogram:TRACe:STARt?**
Instruction	Sets the display trace start. Gets the display trace start.
Parameter Type	Float
Parameter Range	
Return	Float
Default	
Menu	Meas Setup > Ogram View Stop
Example	:DISP:VIEW:SPEC:TRAC:STAR 100 :DISP:VIEW:SPEC:TRAC:STAR?

9.EMI Measurement

9.1 Frequency Subsection

[:SENSe]:FREQuency:CENTer

[:SENSe]:FREQuency:MIDSpan

[:SENSe]:FREQuency:STARt

[:SENSe]:FREQuency:STOP

[:SENSe]:FREQuency:SPAN

:DISPlay:WINDow:TRACe:X[:SCALe]:SPACing

[:SENSe]:FSCan:RANGe

Command Format	**[:SENSe]:FREQuency:CENTer <freq>** **[:SENSe]:FREQuency:CENTer?**
Instruction	Sets the frequency of Meters in the Frequency Scan measurement. Gets the frequency of Meters.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	0 Hz ~ 7.5 GHz
Return	Float, unit: Hz
Default	165 MHz
Menu	Frequency > Freq(Meter)
Example	:FREQuency:CENTer 0.2 GHz :SENSe:FREQuency:CENTer 0.2 GHz

Command Format	**[:SENSe]:FREQuency:MIDSpan <freq>** **[:SENSe]:FREQuency:MIDSpan?**
Instruction	Sets the frequency at the midspan of the EMI Measurement. Gets the frequency at the midspan.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	50 Hz ~ 7.49999995 GHz
Return	Float, unit: Hz
Default	165 MHz
Menu	Frequency > Midspan Freq
Example	:FREQuency:MIDSpan 0.2 GHz :SENSe:FREQuency:MIDSpan 0.2 GHz

Command Format	**[:SENSe]:FREQuency:STARt <freq>** **[:SENSe]:FREQuency:STARt?**
Instruction	Sets the frequency of the EMI Measurement. Gets the frequency.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	0 Hz ~ 7.4999999 GHz
Return	Float, unit: Hz
Default	30 MHz
Menu	Frequency > Start Freq
Example	:FREQuency:STARt 10 MHz

Command Format	**[:SENSe]:FREQuency:STOP <freq>** **[:SENSe]:FREQuency:STOP?**
Instruction	Sets the frequency at the right side of the graticule display. Gets the stop frequency.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	100 Hz ~ 7.5 GHz
Return	Float, unit: Hz
Default	300 MHz
Menu	Frequency > Stop Freq
Example	:FREQuency:STOP 1.0 GHz

Command Format	**[:SENSe]:FREQuency:SPAN <freq>** **[:SENSe]:FREQuency:SPAN?**
Instruction	Sets the span of the EMI Measurement. Gets the span frequency.

Parameter Type	Float, unit: Hz, kHz, MHz, GHz
Parameter Range	100 Hz ~ 7.5 GHz
Return	Float, unit: Hz
Default	270 MHz
Menu	Span > Span
Example	:FREQuency:SPAN 1 GHz

Command Format	**:DISPlay:WINDow:TRACe:X[:SCALe]:SPACing LOGarithmic|LINear** **:DISPlay:WINDow:TRACe:X[:SCALe]:SPACing?**
Instruction	Chooses a linear or logarithmic scaling for the frequency axis.
Parameter Type	Enumeration
Parameter Range	LOGarithmic|LINear
Return	Enumeration
Default	LOG
Menu	Span > X Scale
Example	:DISP:WIND:TRAC:X:SPAC LIN

Command Format	**[:SENSe]:FSCan:RANGe CISA|CISB|CISC|CISBC|CISD**
Instruction	Selects the span range in CISPR standard.
Parameter Type	Enumeration
Parameter Range	CISA|CISB|CISC|CISBC|CISD
Return	Enumeration
Default	CISC
Menu	Span > CISPR Band
Example	:FSC:RANG CISA

9.2 Amplitude Subsection

:DISPlay:FSCan:VIEW:WINDow:TRACe:Y[:SCALe]:RLEVel

[:SENSe]:POWer[:RF]:ATTenuation

[:SENSe]:POWer[:RF]:ATTenuation:AUTO

[:SENSe]:POWer[:RF]:GAIN[:STATe]

:UNIT:POWer

:DISPlay:FSCan:VIEW:WINDow:TRACe:Y[:SCALe]:PDIVision

:DISPlay:WINDow:TRACe:Y:SCALe:RLEVel:OFFSet

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

Command Format	**:DISPlay:FSCan:VIEW:WINDow:TRACe:Y[:SCALe]:RLEVel <value>** **:DISPlay:FSCan:VIEW:WINDow:TRACe:Y[:SCALe]:RLEVel?**
Instruction	This command sets the reference level for the Y-axis. Gets reference level.
Parameter Type	Float, unit: dBm, dBmV, dBuV, V, W
Parameter Range	Unit is dBm: -100 dBm ~ 30 dBm, Unit is dBmV: -53.01 dBmV ~ 76.99 dBmV, Unit is dBuV: 6.99 dBuV ~ 136.99 dBuV, Unit is Volts: 2.24 uV ~ 7.07 V Unit is Watts: 100 fW ~ 1 W.
Return	Float, unit: dBm
Default	0 dBm
Menu	Amplitude > Ref Level
Example	:DISPlay:WINDow:TRACe:Y:RLEVel 20 DBM

Command Format	**[:SENSe]:POWer[:RF]:ATTenuation** **[:SENSe]:POWer[:RF]:ATTenuation?**
Instruction	Sets the input attenuator of the EMI Measurement. Gets the input attenuator.
Parameter Type	Integer
Parameter Range	0 dB ~ 51 dB
Return	Integer, unit: dB
Default	20 dB
Menu	Amplitude > Attenuator
Example	:POWer:ATTenuation 10

Command Format	**[:SENSe]:POWer[:RF]:ATTenuation:AUTO OFF|ON|0|1** **[:SENSe]:POWer[:RF]:ATTenuation:AUTO?**
Instruction	This command turns on/off auto input port attenuator state. Gets input port attenuator state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	ON

Menu	Amplitude > Attenuator
Example	:POWer:ATTenuation:AUTO?

Command Format	**[:SENSe]:POWer[:RF]:GAIN[:STATe] OFF|ON|0|1** **[:SENSe]:POWer[:RF]:GAIN[:STATe]?**
Instruction	Turns the internal preamp on/off. Gets preamp on-off state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	OFF
Menu	Amplitude > Preamp
Example	:POWer:GAIN ON

Command Format	**:UNIT:POWer DBM|DBMV|DBUV|V|W** **:UNIT:POWer?**
Instruction	Specifies amplitude units for the input, output and display. Gets amplitude units.
Parameter Type	Enumeration
Parameter Range	DBM|DBMV|DBUV|DBUA|V|W,
Return	Enumeration
Default	DBUV
Menu	Amplitude > Units
Example	:UNIT:POWer DBMV

Command Format	**:DISPlay:FSCan:VIEW:WINDow:TRACe:Y[:SCALe]:PDIVision <integer>** **:DISPlay:FSCan:VIEW:WINDow:TRACe:Y[:SCALe]:PDIVision?**
Instruction	This command sets the per-division display scaling for the y-axis when scale type of Y axis is set to Log. Gets Scale/Div when scale type of Y axis is set to Log.
Parameter Type	Integer
Parameter Range	1 dB ~ 10 dB
Return	Float, unit: dB
Default	10 dB
Menu	Amplitude > Scale/Div

Example	:DISPlay:WINDow:TRACe:Y:PDIVision 10 dB

Command Format	**:DISPlay:WINDow:TRACe:Y:SCALe:RLEVel:OFFSet <value>** **:DISPlay:WINDow:TRACe:Y:SCALe:RLEVel:OFFSet?**
Instruction	Sets reference offsets. Gets reference offsets.
Parameter Type	Float
Parameter Range	-100 dB ~ 100 dB
Return	Float, unit: dB
Default	0dB
Menu	Amplitude > Ref Offset
Example	:DISPlay:WINDow:TRACe:Y:SCALe:RLEVel:OFFSet 2

Command Format	**:DISPlay:WINDow:TRACe:Y[:SCALe]:SPACing LINear|LOGarithmic** **:DISPlay:WINDow:TRACe:Y[:SCALe]:SPACing?**
Instruction	Toggles the vertical graticule divisions between logarithmic unit and linear unit. The default logarithmic unit is dBm, and the linear unit is V. Gets scale type.
Parameter Type	Enumeration
Parameter Range	LINear|LOGarithmic
Return	Enumeration
Default	LOGarithmic
Menu	Amplitude > Scale Type
Example	:DISPlay:WINDow:TRACe:Y:SPACing LINear

9.3 Sweep Subsection

INITiate[:IMMediate]

:INITiate2:CONTinuous

[:SENSe]:SWEep:COUNt

[:SENSe]:FSCan:SCAN:PRBW

[:SENSe]:FSCan:SCAN:PRBW:AUTO

[:SENSe]:FSCan:SCAN:POINts?

:INITiate:CONTinuous

Command Format	**INITiate[:IMMediate]**
Instruction	Restart the current sweep.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	
Example	:INIT:IMM

Command Format	**:INITiate2:CONTinuous OFF|ON|0|1** **:INITiate2:CONTinuous?**
Instruction	This control determines whether the scan is continuous or a single scan. The scan will not start until you manually initiate the scanning through the Start or Clear List and Start functions.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	ON
Menu	Sweep > Scan Mode
Example	:INITiate2:CONTinuous OFF

Command Format	**[:SENSe]:SWEep:COUNt <integer>** **[:SENSe]:SWEep:COUNt?**
Instruction	Sets sweep numbers, when single sweep on. Gets sweep numbers, when single sweep on.
Parameter Type	Integer
Parameter Range	1 ~ 99999
Return	Integer
Default	1
Menu	Sweep > Numbers
Example	:SWEep:COUNt 10

Command Format	**[:SENSe]:FSCan:SCAN:PRBW < Float >** **[:SENSe]:FSCan:SCAN:PRBW?**
Instruction	Sets RBW/Step. Gets RBW/Step.

Parameter Type	Discrete, Float
Parameter Range	0.1, 0.3, 0.5, 1, 2, 3
Return	Float
Default	1
Menu	Sweep > RBW/Step
Example	:FSCan:SCAN:PRBW 2

Command Format	**[:SENSe]:FSCan:SCAN:PRBW:AUTO < Boolean >** **[:SENSe]:FSCan:SCAN:PRBW:AUTO?**
Instruction	Sets RBW/Step AUTO.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	ON
Menu	Sweep > RBW/Step
Example	:FSCan:SCAN:PRBW:AUTO 1

Command Format	**[:SENSe]:FSCan:SCAN:POINts?**
Instruction	Gets sweep points.
Parameter Type	None
Parameter Range	None
Return	Integer
Default	2251
Menu	Sweep > Sweep Points
Example	:FSC:SCAN:POINts?

Command Format	**:INITiate:CONTinuous OFF|ON|0|1** **:INITiate:CONTinuous?**
Instruction	Sets meter sweep mode. Gets meter sweep mode.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	ON
Menu	Sweep > Meter Mode
Example	:INITiate:CONTinuous OFF

9.4 Bandwidth Subsection

[:SENSe]:FSCan:SCAN:BWIDth[:RESolution]

[:SENSe]:FSCan:SCAN:BWIDth[:RESolution]:AUTO

[:SENSe]:BWIDth[:RESolution

[:SENSe]:BWIDth[:RESolution]:AUTO

Command Format	**[:SENSe]:FSCan:SCAN:BWIDth[:RESolution] <freq>** **[:SENSe]:FSCan:SCAN:BWIDth[:RESolution]?**
Instruction	Specifies the resolution bandwidth of scan. Gets resolution bandwidth of scan.
Parameter Type	Discrete
Parameter Range	100 Hz, 200 Hz, 300 Hz, 1 kHz, 3 kHz, 9 kHz, 10 kHz, 30 kHz, 100 kHz, 120 kHz, 300 kHz, 1 MHz
Return	Float, unit: Hz
Default	120 kHz
Menu	BW > RBW(Scan)
Example	:FSC:SCAN:BWID 9 kHz

Command Format	**[:SENSe]:FSCan:SCAN:BWIDth[:RESolution]:AUTO OFF|ON|0|1** **[:SENSe]:FSCan:SCAN:BWIDth[:RESolution]:AUTO?**
Instruction	Turns on/off auto resolution bandwidth state of scan. Gets resolution bandwidth state of scan.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1

Default	ON
Menu	BW > RBW(Scan)
Example	:FSC:SCAN:BWID:AUTO ON

Command Format	**[:SENSe]:BWIDth[:RESolution] <freq>** **[:SENSe]:BWIDth[:RESolution]?**
Instruction	Specifies the resolution bandwidth of meters. Gets the resolution bandwidth of meters.
Parameter Type	Discrete
Parameter Range	200 Hz, 9 kHz, 120 kHz
Return	Float, unit: Hz
Default	9 kHz
Menu	BW > RBW(Meter)
Example	:BWID 120 kHz

Command Format	**[:SENSe]:BWIDth[:RESolution]:AUTO OFF|ON|0|1** **[:SENSe]:BWIDth[:RESolution]:AUTO?**
Instruction	Turns on/off auto resolution bandwidth state of meters. Gets resolution bandwidth state of meters.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	ON
Menu	BW > RBW(Meter)
Example	:BWID:AUTO ON

9.5 Trace Subsection

:TRACe[1]|2|3:FSCan:TYPE

[:SENSe]:DETector:TRACe[1]|2|3[:FUNCtion]

[:SENSe]:AVERage:TRACe[1]|2|3:COUNt

:TRACe1|2|3:FSCan [:DATA]?

:FORMat[:TRACe][:DATA]

Command Format	**:TRACe[1]|2|3:FSCan:TYPE WRITe|MAXHold|MINHold|VIEW|BLANk|AVERage** **:TRACe[1]|2|3:FSCan:TYPE?**
Instruction	Selects the display mode for the selected trace. Gets the display mode for the selected trace.

Parameter Type	Enumeration
Parameter Range	WRITe: puts the trace in the normal mode, updating the data. MAXHold: displays the highest measured trace value for all the data that has been measured since the function was turned on. MINHold: displays the lowest measured trace value for all the data that has been measured since the function was turned on. VIEW: turns on the trace data so that it can be viewed on the display. BLANk: turns off the trace data so that it is not viewed on the display. AVERage: averages the trace for test period.
Return	Enumeration
Default	Trace1: WRITe, Trace2|3: BLANk
Menu	Trace
Example	TRAC1:FSC:TYPE VIEW

Command Format	**[:SENSe]:DETector:TRACe[1]|2|3[:FUNCtion] POSitive|QPEak|EAVerage** **[:SENSe]:DETector:TRACe[1]|2|3[:FUNCtion]?**
Instruction	Specifies the detection mode. For each trace interval (bucket), average detection displays the average of all the samples within the interval.
Parameter Type	Enumeration
Parameter Range	POSitive|QPEak|EAVerage
Return	Enumeration
Default	Trace 1: POS, Trace 2: QPE, Trace 3: EAV
Menu	Detect
Example	:DET:TRAC2 POS

Command Format	**[:SENSe]:AVERage:TRACe[1]|2|3:COUNt <integer>** **[:SENSe]:AVERage:TRACe[1]|2|3:COUNt?**
Instruction	Specifies the number of measurements that are combined.
Parameter Type	Integer
Parameter Range	1 ~ 999
Return	Integer
Default	1
Menu	Trace > Average
Example	:AVERage:TRACe1:COUNt 10

Command Format	**:TRACe1|2|3:FSCan [:DATA]?**
Instruction	This query command returns the current displayed data.

Parameter Type	None
Parameter Range	None
Return	String
Default	1
Menu	None
Example	:TRACe:DATA?

Command Format	**:FORMat[:TRACe][:DATA] ASCii|REAL** **:FORMat[:TRACe][:DATA]?**
Instruction	Sets trace data type. Gets trace data type.
Parameter Type	Enumeration
Parameter Range	ASCii REAL: single precision floating-point (float)
Return	String
Default	ASCii
Menu	None
Example	:FORMat ASCii

9.6 Marker Subsection

:CALCulate:FSCan:MARKer[1]|2|3|4|5|6:STATe

:CALCulate:FSCan:MARKer[1]|2|3|4|5|6:MODE

:CALCulate:FSCan:MARKer[1]|2|3|4|5|6:TRACe

:CALCulate:FSCan:MARKer[1]|2|3|4|5|6:REFerence

:CALCulate:FSCan:MARKer[1]|2|3|4|5|6:X

:CALCulate:FSCan:MARKer[1]|2|3|4|5|6:Y?

:CALCulate:FSCan:MARKer[1]|2|3|4|5|6[:SET]:SLISt

:CALCulate:FSCan:MARKer[1]|2|3|4|5|6[:SET]:METer

:CALCulate:FSCan:MARKer[1]|2|3|4|5|6:TO:METer:CALCulate:FSCan:MARKer[1]|2|3|4|5
|6:TO:METer**:CALCulate:MARKer:PEAK:THReshold**

:CALCulate:MARKer:PEAK:EXCursion

:CALCulate:FSCan:MARKer[1]|2|3|4|5|6:MAXimum

:CALCulate:MARKer[1]|2|3|4|5|6:MAXimum:LEFT

:CALCulate:MARKer[1]|2|3|4|5|6:MAXimum:RIGHt

Command Format	**:CALCulate:FSCan:MARKer[1]|2|3|4|5|6:STATe OFF|ON|0|1** **:CALCulate:FSCan:MARKer[1]|2|3|4|5|6:STATe?**

Instruction	This command toggles the selected marker status between on and off. Gets marker state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	OFF
Menu	Marker
Example	:CALC:FSC:MARK1:STAT ON

Command Format	**:CALCulate:FSCan:MARKer[1]|2|3|4|5|6:MODE POSition|DELTa|OFF|FIXed** **:CALCulate:FSCan:MARKer[1]|2|3|4|5|6:MODE?**
Instruction	Selects the type of markers that you want to activate. Gets the type of markers.
Parameter Type	Enumeration
Parameter Range	POSition: selects a normal marker that can be positioned on a trace and from which trace information will be generated. DELTa: activates a pair of markers, one of which is fixed at the current marker location. The other marker can then be moved around on the trace. The marker readout shows the marker value which moves. FIXed: Active marker fixed at current position OFF: turns the designated marker off. If a marker is not active when the mode is queried, “off” will be returned.
Return	Enumeration
Default	OFF
Menu	Marker
Example	:CALC:FSC:MARK1:MODE POSition

Command Format	**:CALCulate:FSCan:MARKer[1]|2|3|4|5|6:TRACe 1|2|3** **:CALCulate:FSCan:MARKer[1]|2|3|4|5|6:TRACe?**
Instruction	This command assigns the specified marker to the designated trace 1, 2 or 3. Gets the specified marker to which trace.
Parameter Type	Enumeration
Parameter Range	MARKer:1|2|3|4|5|6 TRACe:1|2|3
Return	Enumeration
Default	1
Menu	Marker > Select Trace
Example	:CALC:FSC:MARK1:TRAC 2

Command Format	**:CALCulate:FSCan:MARKer[1]|2|3|4|5|6:REFerence <integer>** **:CALCulate:FSCan:MARKer[1]|2|3|4|5|6:REFerence?**

Instruction	Sets marker relative to. Gets marker relative to.
Parameter Type	Enumeration
Parameter Range	1|2|3|4|5|6
Return	Enumeration
Default	1
Menu	Marker > Relative To
Example	:CALC:FSC:MARK1:REF 2

Command Format	**:CALCulate:FSCan:MARKer[1]|2|3|4|5|6:X <para> :CALCulate:FSCan:MARKer[1]|2|3|4|5|6:X?**
Instruction	This command positions the designated marker on its assigned trace at the specified trace X value.
Parameter Type	Float, unit: Hz, kHz, MHz, GHz, Default “Hz”
Parameter Range	0 Hz ~ 7.5 GHz
Return	Float, unit: Hz
Default	165 MHz
Menu	Marker > Normal
Example	:CALC:FSC:MARK1:X 0.4 GHz :CALC:FSC:MARK1:X?

Command Format	**:CALCulate:FSCan:MARKer[1]|2|3|4|5|6:Y?**
Instruction	This command reads the current Y value for the designated marker.
Parameter Type	None
Parameter Range	None
Return	Float, unit: dBm
Default	None
Menu	Marker > Normal
Example	:CALC:FSC:MARK1:Y?

Command Format	**:CALCulate:FSCan:MARKer[1]|2|3|4|5|6[:SET]:SLISt**
Instruction	Set marker frequency to signal list.
Parameter Type	None
Parameter Range	None

Return	None
Default	None
Menu	Marker→ > M→List
Example	:CALC:FSC:MARK1:SLIS

Command Format	**:CALCulate:FSCan:MARKer[1]|2|3|4|5|6[:SET]:METer**
Instruction	Replaces the frequency of the selected marker with Frequency (Meters). If the Meters frequency is out of the current span, an error is generated and the marker is not moved.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Marker→ > M→Meter
Example	:CALC:FSC:MARK1:MET

Command Format	**:CALCulate:FSCan:MARKer[1]|2|3|4|5|6:TO:METer**
Instruction	Replaces Frequency (Meters) with the frequency identified by the selected marker.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Marker→ > Meter→M
Example	:CALC:FSC:MARK1:TO:MET

Command Format	**:CALCulate:MARKer:PEAK:THReshold <value>** **:CALCulate:MARKer:PEAK:THReshold?**
Instruction	Specifies the minimum signal level for the analyzers internal peak identification routine to recognize a signal as a peak. This applies to all traces and all windows. Gets the minimum signal level for the analyzers internal peak identification routine to recognize a signal as a peak.
Parameter Type	Float, unit: dBm, dBmV, dBuV, dBuA, V, W
Parameter Range	-200 dBm ~ 200 dBm
Return	Float, unit: dBm
Default	-100 dBm

Menu	Peak > Search Config > Peak Threshold
Example	:CALC:MARK:PEAK:THR -50

Command Format	**:CALCulate:MARKer:PEAK:EXCursion <value>** **:CALCulate:MARKer:PEAK:EXCursion?**
Instruction	Specifies the minimum signal excursion above the threshold for the internal peak identification routine to recognize a signal as a peak.
Parameter Type	Float, unit: dB
Parameter Range	0 ~ 200.0dB
Return	Float, unit: dB
Default	15 dB
Menu	Peak > Search Config > Peak Excursion
Example	:CALC:MARK:PEAK:EXC 10

Command Format	**:CALCulate:FSCan:MARKer[1]|2|3|4|5|6:MAXimum**
Instruction	Performs a peak search based on the search mode settings. (based on the search mode settings, include: peak threshold and peak excursion, Reference Commands: :CALCulate:MARKer:PEAK:THReshold :CALCulate:MARKer:PEAK:EXCursion)
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Peak
Example	:CALCulate:MARKer4:MAXimum

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6:MAXimum:LEFT**
Instruction	Places the selected marker on the next highest signal peak to the left of the current marked peak. (based on the search mode settings, include: peak threshold and peak excursion, Reference Commands: :CALCulate:MARKer:PEAK:THReshold :CALCulate:MARKer:PEAK:EXCursion)
Parameter Type	None
Parameter Range	None
Return	None

Default	None
Menu	Peak > Left Peak
Example	:CALCulate:MARKer1:MAXimum:LEFT

Command Format	**:CALCulate:MARKer[1]|2|3|4|5|6:MAXimum:RIGHt**
Instruction	Places the selected marker on the next highest signal peak to the right of the current marked peak. (based on the search mode settings, include: peak threshold and peak excursion, Reference Commands: :CALCulate:MARKer:PEAK:THReshold :CALCulate:MARKer:PEAK:EXCursion)
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Peak > Right Peak
Example	:CALCulate:MARKer1:MAXimum:RIGHt

9.7 Limit Subsection

:CALCulate:LLINe:TEST

:CALCulate:LLINe[1]|2|3:STATe

:CALCulate:FSCan:LLINe[1]|2|3:MARGin

:CALCulate:FSCan:LLINe[1]|2|3:MARGin:STATe

:CALCulate:FSCan:LLINe[1]|2|3:TRACe

:CALCulate:LLINe[1]|2:MODE

:CALCulate:LLINe[1]|2|3:Y

:CALCulate:LLINe[1]|2|3:DATA

:CALCulate:LLINe[1]|2|3:ADD

:CALCulate:LLINe[1]|2|3:DELete

:CALCulate:LLINe[1]|2|3:ALL:DELete

:CALCulate:LLINe:FAIL?

Command Format	**:CALCulate:LLINe:TEST OFF|ON|0|1** **:CALCulate:LLINe:TEST?**
Instruction	Sets limit test start or stop.
Parameter Type	Boolean

Parameter Range	OFF|ON|0|1
Return	0|1
Default	0
Menu	Limit > Test
Example	:CALCulate:LLINe:TEST ON

Command Format	**:CALCulate:LLINe[1]|2|3:STATe OFF|ON|0|1** **:CALCulate:LLINe[1]|2|3:STATe?**
Instruction	Sets limit line state. Gets limit line state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	OFF
Menu	Limit > Limit1|2|3
Example	:CALCulate:LLINe1:STATe OFF

Command Format	**:CALCulate:FSCan:LLINe[1]|2|3:MARGin <value>** **:CALCulate:FSCan:LLINe[1]|2|3:MARGin?**
Instruction	Sets limit margin value. Gets limit margin value.
Parameter Type	Float, units: dB
Parameter Range	-40 dB ~ 0 dB
Return	Float, units: dB
Default	-6 dB
Menu	Limit > Limit1|2|3 > Margin
Example	:CALC:FSC:LLIN1:MARG 0 dB

Command Format	**:CALCulate:FSCan:LLINe[1]|2|3:MARGin:STATe OFF|ON|0|1** **:CALCulate:FSCan:LLINe[1]|2|3:MARGin:STATe?**
Instruction	Sets limit margin state. Gets limit margin state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	0

Menu	Limit > Limit1|2|3 > Margin
Example	:CALC:FSC:LLIN1:MARG:STAT OFF

Command Format	**:CALCulate:FSCan:LLINe[1]|2|3:TRACe 1|2| 3** **:CALCulate:FSCan:LLINe[1]|2|3:TRACe?**
Instruction	The Trace column selects the trace that you want the limit to test.
Parameter Type	Enumeration
Parameter Range	1|2|3
Return	Enumeration
Default	1
Menu	Limit > Limit1|2|3 Setup > Test Trace
Example	:CALC:FSC:LLIN1:TRAC 2

Command Format	**:CALCulate:LLINe[1]|2:MODE LINE|POINt** **:CALCulate:LLINe[1]|2:MODE?**
Instruction	Sets limit mode. Gets limit mode
Parameter Type	Enumeration
Parameter Range	LINE|POINt
Return	Enumeration
Default	LINE
Menu	Limit > Limit1|2 Edit > Mode
Example	:CALC:LLIN1: MODE POINt

Command Format	**:CALCulate:LLINe[1]|2|3:Y <value>** **:CALCulate:LLINe[1]|2|3:Y?**
Instruction	Sets the Y-axis value of a limit line. Limit line Y-axis value is set independently and is not affected by the X-axis units. Gets the Y-axis value of a limit line.
Parameter Type	Float, units: dBm, dBmV, dBuV, dBuA, V, W
Parameter Range	
Return	Float
Default	0
Menu	Limit > Limit1|2|3 Setup > Edit > Amplitude
Example	:CALC:LLIN1:Y 5 dBm

Command Format	:CALCulate:LLINe[1]|2|3:DATA <x-axis>,<ampl>{,<x-axis>, <ampl>} :CALCulate:LLINe[1]|2|3:DATA?
Instruction	Uses this command to define the limit points. Gets the defined limit points.
Parameter Type	X-axis: Float, unit: Hz, kHz, MHz, GHz Amplitude: Float, unit: dBm, dBmV, dBuV, dBuA, V, W
Parameter Range	X-axis: 0 ~ 7.5 GHz Amplitude:
Return	X-axis: Float Amplitude: Float
Default	X-axis: -1 Hz Amplitude:
Menu	Limit > Limit1|2|3 Edit
Example	:CALC:LLIN1:DATA 10000000,-20,20000000,-30

Command Format	:CALCulate:LLINe[1]|2|3:ADD <x-axis>,<ampl>
Instruction	Adds limit point data.
Parameter Type	X-axis: Float Amplitude: Float
Parameter Range	X-axis: 0 ~ 7.5 GHz Amplitude: None
Return	X-axis: Float Amplitude: Float
Default	X-axis: -1 Hz Amplitude: 0 dBm
Menu	Limit > Limit1|2|3 Edit
Example	:CALC:LLIN1:ADD 10000000,-20

Command Format	:CALCulate:LLINe[1]|2|3:DELete <number>
Instruction	Uses this command to delete the assigned limit point.
Parameter Type	Integer
Parameter Range	None
Return	None
Default	None
Menu	Limit > Limit1|2|3 Edit > Del Point
Example	:CALC:LLIN1:DEL 2

Command Format	:CALCulate:LLINe[1]|2|3:ALL:DELete
Instruction	Uses this command to define all the limits points.

Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Limit > Limit1|2|3 Edit > Del All
Example	:CALC:LLIN2:ALL:DELete

Command Format	**:CALCulate:LLINe:FAIL?**
Instruction	This query command returns the limits pass/failed result. If the test result fails, this command will get result FAIL. If the test result passes, it will get result PASS.
Parameter Type	None
Parameter Range	None
Return	PASS|FAIL
Default	None
Menu	None
Example	:CALC:LLIN:FAIL?

9.8 Measurement Subsystem

[:SENSe]:FSCan:SEQuence

[:SENSe]:FSCan:SCAN:TIME

:DISPlay:METer[1]|2|3[:STATe]

[:SENSe]:METer[1]|2|3:DETector:DWELl [:SENSe]:METer[1]|2|3:DETector

:CALCulate:METer[1]|2|3:LIMit[:DATA]

:CALCulate:METer[1]|2|3:LIMit:STATe

:CALCulate:METer[1]|2|3:LIMit:ULLine

:CALCulate:SLISt:MARK:SIGNal

:CALCulate:SLISt:MARK:CLEar:SIGNal

:CALCulate:SLISt:MARK:ALL

:CALCulate:SLISt:MARK:CLEar:ALL

:CALCulate:SLISt:DELete:SIGNal

:CALCulate:SLISt:DELete:ALL

:CALCulate:SLISt:SORT:TYPE

:CALCulate:SLISt:SORT:ORDer

Command Format	**[:SENSe]:FSCan:SEQuence SCAN|SEARch|SSAMeasure|SASearch|SAMeasure|REMeasure** **[:SENSe]:FSCan:SEQuence?**
Instruction	Sets the sequence. Gets the sequence.
Parameter Type	Enumeration
Parameter Range	SCAN | SEARch | SSAMeasure | SASearch | SAMeasure | REMeasure
Return	Enumeration
Default	SCAN
Menu	Meas > Sequence
Example	:FSC:SEQ SEAR

Command Format	**[:SENSe]:FSCan:SCAN:TIME <time>** **[:SENSe]:FSCan:SCAN:TIME?**
Instruction	Sets scan dwell time. Gets scan dwell time.
Parameter Type	Float, unit: ks, s, ms, us
Parameter Range	1 ms ~ 10 s
Return	Float, unit: s
Default	
Menu	Meas > Scan Config > Dwell Time
Example	:FSC:SCAN:TIME 100 ms

Command Format	**:DISPlay:METer[1]|2|3[:STATe] OFF|ON|0|1** **:DISPlay:METer[1]|2|3[:STATe]?**
Instruction	Sets meter state. Gets meter state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1
Return	0|1
Default	1
Menu	Meas > Meter Config > Meter 1|2|3
Example	:DISP:MET1 OFF

Command Format	**[:SENSe]:METer[1]|2|3:DETector:DWELl <time>** **[:SENSe]:METer[1]|2|3:DETector:DWELl?**
Instruction	Sets meter dwell time. Gets meter dwell time.

Parameter Type	Float, unit: ks, s, ms, us
Parameter Range	1 ms ~ 10 s
Return	Float, unit: s
Default	0.01
Menu	Meas > Meter Config > Meter 1|2|3 > Dwell Time
Example	MET1:DET:DWEL 50 ms

Command Format	**[:SENSe]:METer[1]|2|3:DETector POSitive|QPEak|EAVerage** **[:SENSe]:METer[1]|2|3:DETector?**
Instruction	Sets meter detector. Gets meter detector.
Parameter Type	Enumeration
Parameter Range	POSitive|QPEak|EAVerage
Return	Enumeration
Default	Meter1: POS Meter2: QPE Meter3: EAV
Menu	Meas > Meter Config > Meter 1|2|3 Detector
Example	MET1:DET EAV

Command Format	**:CALCulate:METer[1]|2|3:LIMit[:DATA] <ampl>** **:CALCulate:METer[1]|2|3:LIMit[:DATA]?**
Instruction	Sets limit value of meter. Gets limit value of meter.
Parameter Type	Float, unit: dBm, dBmV, dBuV, dBuA, V, W
Parameter Range	
Return	Float
Default	0
Menu	Meas > Meter Config > Meter1|2|3 Limit > Value
Example	:CALC:MET1:LIM 20 dBuV

Command Format	**:CALCulate:METer[1]|2|3:LIMit:STATe OFF|ON|0|1** **:CALCulate:METer[1]|2|3:LIMit:STATe?**
Instruction	Sets meter limit state. Gets meter limit state.
Parameter Type	Boolean
Parameter Range	OFF|ON|0|1

Return	0|1
Default	0
Menu	Meas > Meter Config > Meter1|2|3 Limit > Limit
Example	:CALC:MET1:LIM:STAT ON

Command Format	**:CALCulate:METer[1]|2|3:LIMit:ULLine LIMit1 | LIMit2 | LIMit3**
Instruction	Sets limit to meter limit value, You can't abbreviate the parameter as LIM1.
Parameter Type	Enumeration
Parameter Range	
Return	
Default	
Menu	Meas > Meter Config > Meter Limit
Example	:CALCulate:METer1:LIMit:ULLine LIMit1

Command Format	**:CALCulate:SLISt:MARK:SIGNal <integer>**
Instruction	Marks the selected signal in signal list.
Parameter Type	Integer
Parameter Range	
Return	None
Default	None
Menu	Meas > List Operation > Mark Signal
Example	:CALC:SLIS:MARK:SIGN 1

Command Format	**:CALCulate:SLISt:MARK:CLEar:SIGNal <integer>**
Instruction	Clears the mark from the selected signal.
Parameter Type	Integer
Parameter Range	
Return	None
Default	None
Menu	Meas > List Operation > Clear Mark
Example	:CALC:SLIS:MARK:CLE:SIGN 1

Command Format	**:CALCulate:SLISt:MARK:ALL**
Instruction	Marks all signals in signal list.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Meas > Mark All
Example	:CALC:SLIS:MARK:ALL

Command Format	**:CALCulate:SLISt:MARK:CLEar:ALL**
Instruction	Clears all the marks from the signal list.
Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Meas > List Operation > Clear All Marks
Example	:CALC:SLIS:MARK:CLE:ALL

Command Format	**:CALCulate:SLISt:DELete:SIGNal <integer>**
Instruction	Deletes the selected signal.
Parameter Type	Integer
Parameter Range	
Return	None
Default	None
Menu	Meas > List Operation > Delete Signal
Example	:CALC:SLIS:DEL:SIG 1

Command Format	**:CALCulate:SLISt:DELete:ALL**
Instruction	Deletes all signals from signal list.

Parameter Type	None
Parameter Range	None
Return	None
Default	None
Menu	Meas > List Operation > Delete All
Example	:CALC:SLIS:ALL

Command Format	**:CALCulate:SLISt:SORT:TYPE FREQuency | DAMPlitude | DLDelta | TIME**
Instruction	DAMPlitude and DLDelta corresponds to Det.1 and Det.1△
Parameter Type	Enumeration
Parameter Range	
Return	None
Default	None
Menu	Meas > List Operation > Sort By
Example	:CALC:SLIS:SORT:TYPE FREQ

Command Format	**:CALCulate:SLISt:SORT:ORDer ASCending|DESCending** **:CALCulate:SLISt:SORT:ORDer?**
Instruction	Sets the sort order of signal list. Gets the sort order of signal list.
Parameter Type	Enumeration
Parameter Range	ASCending|DESCending
Return	ASC|DESC
Default	ASC
Menu	Meas > List Operation > Sort Order
Example	:CALC:SLIS:SORT:ORD DESC

10.Programming Examples

This chapter gives some examples for the programmer. In these examples you can see how to use the VISA or sockets, in combination with the commands have been described above to control the spectrum analyzer. By following these examples, you can develop many more applications.

10.1 Examples of Using VISA

10.1.1 Example of VC++

Environment: Win7 32bit system, Visual Studio

The functions of this example: use the NI-VISA, to control the device with USBTMC or TCP/IP access to do a write and read.

Follow the steps to finish the example:

1、Open Visual Studio, create a new VC++ win32 console project.

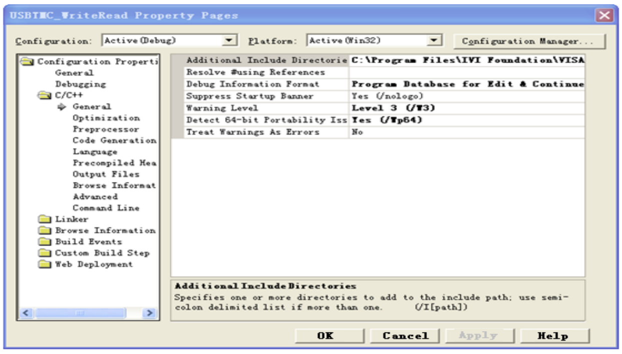
2、Set the project environment to use the NI-VISA lib, there are two ways to use NI-VISA, static or automatic:

1)Static: find files: visa.h, visatype.h, visa32.lib in NI-VISA install path. Copy them to your project, and add them into project. In the projectname.cpp file, add the follow two lines:

```
#include "visa.h"

#pragma comment(lib,"visa32.lib")
```

2)Automatic:

Set the .h file include directory, the NI-VISA install path, in our computer we set the path is: C:\Program Files\IVI Foundation \VISA\WinNT\include. Set this path to project---properties---c/c++---General---Additional Include Directories: See the picture.



Additional Include Directorie	C:\Program Files\IVI Foundation\VISA
Resolve #using References	
Debug Information Format	Program Database for Edit & Continue
Suppress Startup Banner	Yes (/nologo)
Warning Level	Level 3 (/W3)
Detect 64-bit Portability Iss	Yes (/Wp64)
Treat Warnings As Errors	No

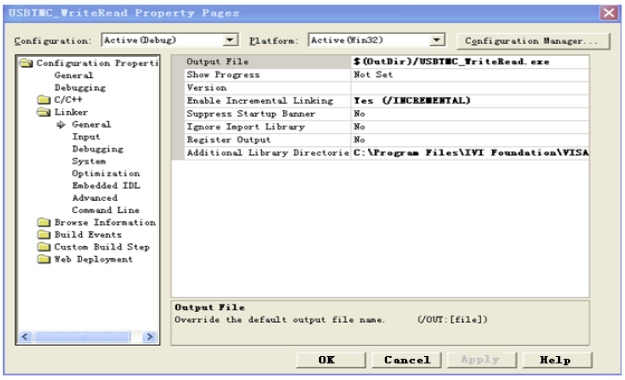
Additional Include Directories

Specifies one or more directories to add to the include path; use semi-colon delimited list if more than one. (/I[path])

Set lib path set lib file:

Set lib path: the NI-VISA install path, in our computer we set the path is: C:\Program Files\IVI Foundation\VISA\WinNT

\lib\msc. Set this path to project---properties---Linker---General---Additional Library Directories: as seen in the pictures below.



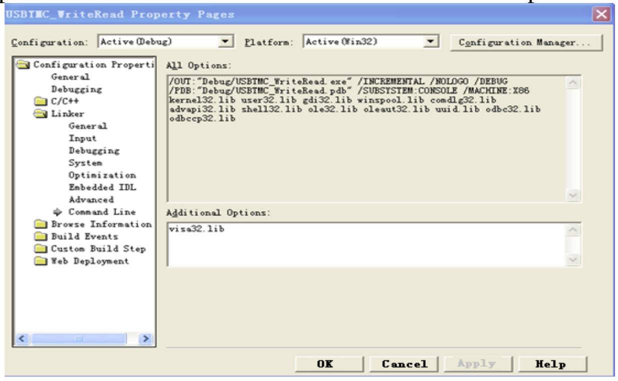
USBTMC_WriteRead Property Pages

Configuration: Active(Debug) Platform: Active(Win32)

Output File	$(OutDir)/USBTMC_WriteRead.exe
Show Progress	Not Set
Version	
Enable Incremental Linking	Yes (/INCREMENTAL)
Suppress Startup Banner	No
Ignore Import Library	No
Register Output	No
Additional Library Directorie	C:\Program Files\IVI Foundation\VISA

Output File
Override the default output file name. (/OUT:[file])

Set lib file:project---properties---Linker---Command Line---Additional Options: visa32.lib



USBTMC_WriteRead Property Pages

Configuration: Active(Debug) Platform: Active(Win32)

All Options:

```
/OUT:"Debug/USBTMC_WriteRead.exe" /INCREMENTAL /NOLOGO /DEBUG
/PDB:"Debug/USBTMC_WriteRead.pdb" /SUBSYSTEM:CONSOLE /MACHINE:X86
kernel32.lib user32.lib gdi32.lib winspool.lib comdlg32.lib
advapi32.lib shell32.lib ole32.lib oleaut32.lib uuid.lib odbc32.lib
odbccp32.lib
```

Additional Options:

```
visa32.lib
```

Include visa.h file: In the projectname.cpp file:

```
#include <visa.h>
```

3、Add codes:

1)USBTMC access code:

Write a function Usbtmc_test:

```
int Usbtmc_test()
{
/* This code demonstrates sending synchronous read & write commands */
/* to an USB Test & Measurement Class (USBTMC) instrument using  */
/* NI-VISA       */
/* The example writes the "*IDN?\n" string to all the USBTMC  */
/* devices connected to the system and attempts to read back  */
/* results using the write and read functions.      */
/* The general flow of the code is */
/* Open Resource Manager   */
/* Open VISA Session to an Instrument                    */
/* Write the Identification Query Using viPrintf   */
/* Try to Read a Response With viScanf    */
/* Close the VISA Session   */
/*********************************************************/
ViSessiondefaultRM;
ViSessioninstr;
ViUInt32numInstrs;
ViFindListfindList;
ViStatus  status;
char instrResourceString[VI_FIND_BUFLEN];
unsignedchar buffer[100];
int i;
/** First we must call viOpenDefaultRM to get the manager
* handle. We will store this handle in defaultRM.*/
status=viOpenDefaultRM (&defaultRM);
if (status<VI_SUCCESS)
{
printf ("Could not open a session to the VISA Resource Manager!\n");
returnstatus;
}
```

```
  /* Find all the USB TMC VISA resources in our system and store the number of resources in the system in
numInstrs.*/
  status = viFindRsrc (defaultRM, "USB?*INSTR", &findList, &numInstrs, instrResourceString);
  if (status<VI_SUCCESS)
  {
  printf ("An error occurred while finding resources.\nPress 'Enter' to continue.");
  fflush(stdin);
  getchar();
  viClose (defaultRM);
  returnstatus;
  }
  /** Now we will open VISA sessions to all USB TMC instruments.
  * We must use the handle from viOpenDefaultRM and we must
  * also use a string that indicates which instrument to open. This
  * is called the instrument descriptor. The format for this string
  * can be found in the function panel by right clicking on the
  * descriptor parameter. After opening a session to the
  * device, we will get a handle to the instrument which we
  * will use in later VISA functions. The AccessMode and Timeout
  * parameters in this function are reserved for future
  * functionality. These two parameters are given the value VI_NULL.*/
  for (i=0; i<int(numInstrs); i++)
  {
  if (i> 0)
  { viFindNext (findList, instrResourceString);
  } status = viOpen (defaultRM, instrResourceString, VI_NULL, VI_NULL, &instr);
  if (status<VI_SUCCESS)
  {
   printf ("Cannot open a session to the device %d.\n", i+1);
   continue;
  }
  /* * At this point we now have a session open to the USB TMC instrument.
  * We will now use the viPrintf function to send the device the string "*IDN?\n",
  * asking for the device's identification. */
  char * cmmand ="*IDN?\n";
  status = viPrintf (instr, cmmand);
  if (status<VI_SUCCESS)
  {
   printf ("Error writing to the device %d.\n", i+1);
   status = viClose (instr);
   continue;
  }
  /** Now we will attempt to read back a response from the device to
  * the identification query that was sent. We will use the viScanf
  * function to acquire the data.
  * After the data has been read the response is displayed.*/
  status = viScanf(instr, "%t", buffer);
  if (status<VI_SUCCESS)
  { printf ("Error reading a response from the device %d.\n", i+1);
  } else
  { printf ("\nDevice %d: %s\n", i+1, buffer);
  }status = viClose (instr);
  }
  /** Now we will close the session to the instrument using
  * viClose. This operation frees all system resources.      */
  status = viClose (defaultRM);
printf("Press 'Enter' to exit.");
fflush(stdin);
  getchar();return 0;
  }

int _tmain(int argc, _TCHAR* argv[])
{
Usbtmc_test();
return 0;
}
```

2)TCP/IP access code:

Write a function TCP_IP_Test:

```
int TCP_IP_Test(char *pIP)
{
char outputBuffer[VI_FIND_BUFLEN];
ViSessiondefaultRM, instr;
ViStatusstatus;

/* First we will need to open the default resource manager. */
status = viOpenDefaultRM (&defaultRM);
if (status<VI_SUCCESS)
{
printf("Could not open a session to the VISA Resource Manager!\n");
}
/* Now we will open a session via TCP/IP device */
char head[256] ="TCPIP0::";
char tail[] ="::INSTR";

strcat(head,pIP);
strcat(head,tail);
status = viOpen (defaultRM, head, VI_LOAD_CONFIG, VI_NULL, &instr);
if (status<VI_SUCCESS)
{
printf（″An error occurred opening the session\n");
viClose(defaultRM);
}
status = viPrintf(instr, "*idn?\n");
status = viScanf(instr, "%t", outputBuffer);
if (status<VI_SUCCESS)
{
printf("viRead failed with error code: %x \n",status);
viClose(defaultRM);
}else
｛printf ("\nMesseage read from device: %*s\n", 0,outputBuffer);
｝status = viClose (instr);
status = viClose (defaultRM);
printf("Press 'Enter' to exit.");
fflush(stdin);
getchar();return 0;
}
int _tmain(int argc, _TCHAR* argv[])
{
printf("Please input IP address:");
char ip[256];
fflush(stdin);
gets(ip);
TCP_IP_Test(ip);
return 0;
}
```

10.1.2 Example of VB

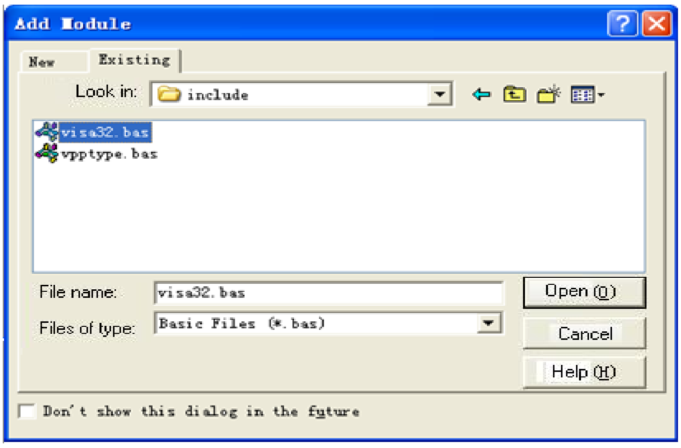
Environment: Win7 32bit system, Microsoft Visual Basic 6.0

The function of this example: Use the NI-VISA, to control the device with USBTMC and TCP/IP access to do a write and read.

Follow the steps to complete the example:

1、Open Visual Basic, build a standard application program project (Standard EXE)

2、Set the project environment to use the NI-VISA lib, Click the Existing tab of Project>>Add Existing Item. Search for the visa32.bas file in the include folder under the NI-VISA installation path and add the file.



Look in: include

File name: visa32.bas

Files of type: Basic Files (*.bas)

This allows the VISA functions and VISA data types to be used in a program.

3、 Add codes:

1)USBTMC access code:

Write a function Usbtmc_test:

```
Private Function Usbtmc_test() As Long

' This code demonstrates sending synchronous read & write commands
 ' to an USB Test & Measurement Class (USBTMC) instrument using
 ' NI-VISA
 ' The example writes the "*IDN?\n" string to all the USBTMC
 ' devices connected to the system and attempts to read back
 ' results using the write and read functions.
 ' The general flow of the code is
 ' Open Resource Manager
 ' Open VISA Session to an Instrument
 ' Write the Identification Query Using viWrite
 ' Try to Read a Response With viRead
 ' Close the VISA Session

Const MAX_CNT = 200

Dim defaultRM As Long
Dim instrsesn As Long
Dim numlnstrs As Long
Dim findList As Long
Dim retCount As Long

Dim status As Long
Dim instrResourceString As String * VI_FIND_BUFLEN
Dim Buffer As String * MAX_CNT
Dim i As Integer

 ' First we must call viOpenDefaultRM to get the manager
 ' handle. We will store this handle in defaultRM.

 status = viOpenDefaultRM(defaultRM)
If (status < VI_SUCCESS) Then
       resultTxt.Text = "Could not open a session to the VISA Resource Manager!"
  Usbtmc_test = status
       Exit Function
End If

 ' Find all the USB TMC VISA resources in our system and store the
 ' number of resources in the system in numlnstrs.

 status = viFindRsrc(defaultRM, "USB?*INSTR", findList, numlnstrs, instrResourceString)
If (status < VI_SUCCESS) Then
```

```
        resultTxt.Text = "An error occurred while finding resources."
  viClose (defaultRM)
  Usbtmc_test = status
Exit Function
End If

 ' Now we will open VISA sessions to all USB TMC instruments.
 ' We must use the handle from viOpenDefaultRM and we must
 ' also use a string that indicates which instrument to open. This
 ' is called the instrument descriptor. The format for this string
 ' can be found in the function panel by right clicking on the
 ' descriptor parameter. After opening a session to the
 ' device, we will get a handle to the instrument which we
 ' will use in later VISA functions. The AccessMode and Timeout
 ' parameters in this function are reserved for future
 ' functionality. These two parameters are given the value VI_NULL.

For i = 0 To numInstrs
If (i > 0) Then
   status = viFindNext(findList, instrResourceString)
End If
  status = viOpen(defaultRM, instrResourceString, VI_NULL, VI_NULL, instrsesn)
If (status < VI_SUCCESS) Then
            resultTxt.Text = "Cannot open a session to the device " + CStr(i + 1)
   GoTo NextFind
End If

  ' At this point we now have a session open to the USB TMC instrument.
  ' We will now use the viWrite function to send the device the string "*IDN?",
  ' asking for the device's identification.

status = viWrite(instrsesn, "*IDN?", 5, retCount)
If (status < VI_SUCCESS) Then
            resultTxt.Text = "Error writing to the device."
   status = viClose(instrsesn)
   GoTo NextFind
End If

  ' Now we will attempt to read back a response from the device to
  ' the identification query that was sent. We will use the viRead
  ' function to acquire the data.
  ' After the data has been read the response is displayed.
  status = viRead(instrsesn, Buffer, MAX_CNT, retCount)
If (status < VI_SUCCESS) Then
            resultTxt.Text = "Error reading a response from the device." + CStr(i + 1)
Else
            resultTxt.Text = "Read from device: " + CStr(i + 1) + " " + Buffer
End If
  status = viClose(instrsesn)

Next i

 ' Now we will close the session to the instrument using
 ' viClose. This operation frees all system resources.
 status = viClose(defaultRM)
 Usbtmc_test = 0
End Function
```

2)TCP/IP access code:

Write a function TCP_IP_Test:

```
Private Function TCP_IP_Test(ByVal ip As String) As Long
Dim outputBuffer As String * VI_FIND_BUFLEN
Dim defaultRM As Long
Dim instrsesn As Long
Dim status As Long
Dim count As Long
```

```
  ' First we will need to open the default resource manager.
  status = viOpenDefaultRM (defaultRM)
  If (status < VI_SUCCESS) Then
        resultTxt.Text = "Could not open a session to the VISA Resource Manager!"
    TCP_IP_Test = status
    Exit Function
  End If

  ' Now we will open a session via TCP/IP device
   status = viOpen(defaultRM, "TCPIP0::" + ip + "::INSTR", VI_LOAD_CONFIG, VI_NULL, instrsesn)
   If (status < VI_SUCCESS) Then
        resultTxt.Text = "An error occurred opening the session"
    viClose (defaultRM)
    TCP_IP_Test = status
  Exit Function
  End If

   status = viWrite(instrsesn, "*IDN?", 5, count)
   If (status < VI_SUCCESS) Then
        resultTxt.Text = "Error writing to the device."
  End If
   status = viRead(instrsesn, outputBuffer, VI_FIND_BUFLEN, count)
  If (status < VI_SUCCESS) Then
        resultTxt.Text = "Error reading a response from the device." + CStr(i + 1)
  Else
        resultTxt.Text = "read from device:" + outputBuffer
  End If
   status = viClose(instrsesn)
   status = viClose(defaultRM)
   TCP_IP_Test = 0
End Function
```

3) Button control code:

```
Private Sub exitBtn_Click()
    End
End Sub
Private Sub tcpipBtn_Click()
    Dim stat As Long
    stat = TCP_IP_Test(ipTxt.Text)
    If (stat < VI_SUCCESS) Then
        resultTxt.Text = Hex(stat)
    End If
End Sub
Private Sub usbBtn_Click()
    Dim stat As Long
    stat = Usbtmc_test
    If (stat < VI_SUCCESS) Then
        resultTxt.Text = Hex(stat)
    End If
End Sub
```

10.1.3 Example of MATLAB

Environment: Win7 32bit system, MATLAB R2013a

The function of this example: Use the NI-VISA, to control the device with USBTMC or TCP/IP access to do a write and read.

Follow the steps to complete the example:

1、Open MATLAB, modify the **current directory**. In this demo, the current directory is modified to D:\USBTMC_TCPIP_Demo.

2、Click **File>>New>>Script** in the Matlab interface to create an empty M file

3、Add codes:

1)USBTMC access code :

Write a function Usbtmc_test.

```
function USBTMC_test()
% This code demonstrates sending synchronous read & write commands
% to an USB Test & Measurement Class (USBTMC) instrument using
% NI-VISA

%Create a VISA-USB object connected to a USB instrument
vu = visa('ni','USB0::0xF4ED::0xEE3A::sdg2000x::INSTR');

%Open the VISA object created
fopen(vu);

%Send the string "*IDN?",asking for the device's identification.
fprintf(vu,'*IDN?');

%Request the data
outputbuffer = fscanf(vu);
disp(outputbuffer);

%Close the VISA object
fclose(vu);
delete(vu);
clear vu;

end
```

2)TCP/IP access code:

Write a function TCP_IP_Test:

```
function TCP_IP_test()
% This code demonstrates sending synchronous read & write commands
% to an TCP/IP instrument using NI-VISA

%Create a VISA-TCPIP object connected to an instrument
%configured with IP address.
vt = visa('ni',['TCPIP0::','10.11.13.32','::INSTR']);

%Open the VISA object created
fopen(vt);

%Send the string "*IDN?",asking for the device's identification.
fprintf(vt,'*IDN?');

%Request the data
outputbuffer = fscanf(vt);
disp(outputbuffer);

%Close the VISA object
fclose(vt);
delete(vt);
clear vt;

end
```

10.1.4 Example of LabVIEW

Environment: Win7 32bit system, LabVIEW 2011

The functions of this example: use the NI-VISA, to control the device with USBTMC and TCP/IP access to do a write and read.

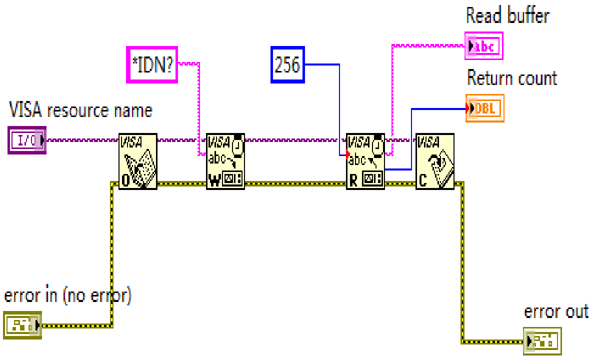
Follow the steps to complete the example:

1、Open LabVIEW, create a VI file.

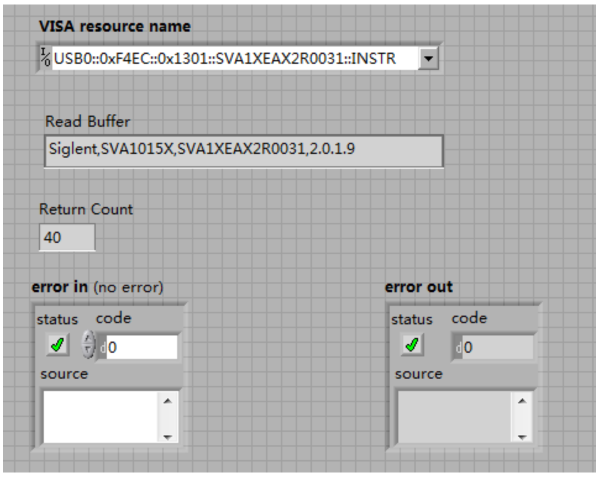
2、Add controls. Right-click in the **Front Panel** interface, select and add **VISA resource name**, error in, error out and some indicators from the Controls column.

3、Open the **Block Diagram** interface. Right-click on the **VISA resource name** and you can select and add the following functions from VISA Palette from the pop-up menu: **VISA Write**, **VISA Read**, **VISA Open** and **VISA Close**.

4、Connect them as shown in the figure below



5、Select the device resource from the VISA Resource Name list box and run the program.



VISA resource name
USB0::0xF4EC::0x1301::SVA1XEAX2R0031::INSTR

Read Buffer
Siglent,SVA1015X,SVA1XEAX2R0031,2.0.1.9

Return Count
40

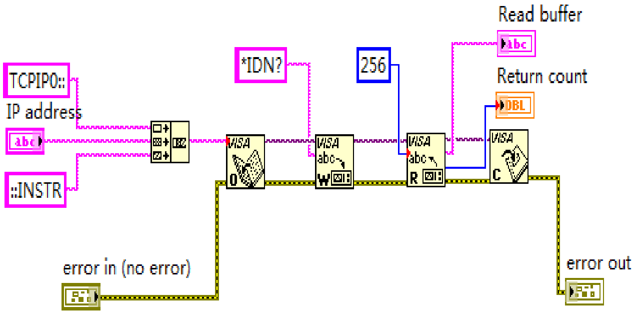
error in (no error)
status ✓ code 0
source

error out
status ✓ code 0
source

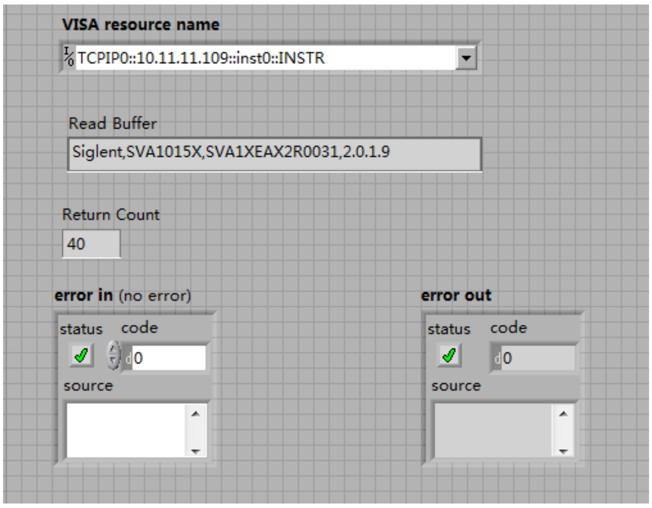
In this example, the VI opens a VISA session to a USBTMC device, writes a command to the device, and reads back the response. In this example, the specific command being sent is the device ID query. Check with your device manufacturer for the device command set. After all communication is complete, the VI closes the VISA session.

6、Communicating with the device via TCP/IP is similar to USBTMC. But you need to change VISA Write and VISA Read Function to Synchronous I/O. The LabVIEW default is asynchronous I/O. Right-click the node and select Synchronous I/O Mod>>Synchronous from the shortcut menu to write or read data synchronously.

7、Connect them as shown in the figure below



8、Input the IP address and run the program.



VISA resource name

TCPIP0::10.11.11.109::inst0::INSTR

Read Buffer

Siglent,SVA1015X,SVA1XEAX2R0031,2.0.1.9

Return Count

40

error in (no error)		error out	
status	code	status	code
✓	0	✓	0
source		source	

10.2 Examples of Using Sockets/Telnet

10.2.1 Example of Python

Python is an interpreted programming language that lets you work quickly and is very portable.Python has a low-level networking module that provides access to the socket interface.Python scripts can be written for sockets to do a variety of test and measurements tasks.

Environment: Win7 32bit system, Python v2.7.5

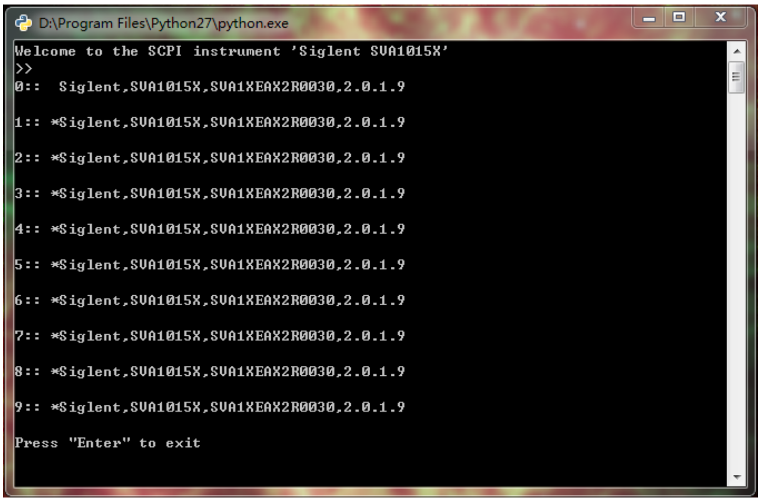
The functions of this example: Open a socket, sends a query, and closes the socket. It does this loop 10 times.

Below is the code of the script:

```
#!/usr/bin/env python
#-*- coding:utf-8 –*-
#-------------------------------------------------------------------------------
# The short script is a example that open a socket, sends a query,
# print the return message and closes the socket.
#-------------------------------------------------------------------------------
import socket   # for sockets
import sys  # for exit
import time # for sleep
#-------------------------------------------------------------------------------
   remote_ip = "10.11.13.32"  # should match the instrument’s IP address
   port = 5025 # the port number of the instrument service
  count = 0

def SocketConnect():
    try:
        #create an AF_INET, STREAM socket (TCP)
        s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    except socket.error:
        print ('Failed to create socket.')
        sys.exit();
    try:
        #Connect to remote server
        s.connect((remote_ip , port))
        info = s.recv(4096)
        print (info)
    except socket.error:
        print ('failed to connect to ip ' + remote_ip)
    return s

def SocketQuery(Sock, cmd):
    try :
        #Send cmd string
      Sock.sendall(cmd)
        Sock.sendall(b'\n')
        time.sleep(1)
    except socket.error:
        #Send failed
        print ('Send failed')
        sys.exit()
    reply = Sock.recv(4096)
    return reply

def SocketClose(Sock):
    #close the socket
    Sock.close()
    time.sleep(.300)
```

```
def main():
    global remote_ip
    global port
    global count

    # Body: send the SCPI commands *IDN? 10 times and print the return message
    s = SocketConnect()
    for i in range(10):
        qStr = SocketQuery(s, b'*IDN?')
        print (str(count) + ":: " + str(qStr))
        count = count + 1     SocketClose(s)
    input('Press "Enter" to exit')

if __name__ == '__main__':
    proc = main()
```

Run result:



```
Welcome to the SCPI instrument 'Siglent SVA1015X'
>>
0::  Siglent,SVA1015X,SVA1XEAX2R0030,2.0.1.9

1:: *Siglent,SVA1015X,SVA1XEAX2R0030,2.0.1.9

2:: *Siglent,SVA1015X,SVA1XEAX2R0030,2.0.1.9

3:: *Siglent,SVA1015X,SVA1XEAX2R0030,2.0.1.9

4:: *Siglent,SVA1015X,SVA1XEAX2R0030,2.0.1.9

5:: *Siglent,SVA1015X,SVA1XEAX2R0030,2.0.1.9

6:: *Siglent,SVA1015X,SVA1XEAX2R0030,2.0.1.9

7:: *Siglent,SVA1015X,SVA1XEAX2R0030,2.0.1.9

8:: *Siglent,SVA1015X,SVA1XEAX2R0030,2.0.1.9

9:: *Siglent,SVA1015X,SVA1XEAX2R0030,2.0.1.9

Press "Enter" to exit
```